



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#) |**Navigate****Index**[Top IT Tasks](#) | [Select from this list](#)[Search TechNet](#)**Navigate by Product**

[Application Center](#)
[BizTalk Server](#)
[Commerce Server](#)
[Exchange Server](#)
[Host Integration Server](#)
[Internet Security & Acceleration Server](#)
[Office](#)
[Site Server Commerce](#)
[Small Business Server](#)
[SQL Server](#)
[Systems Management Server](#)
[Visio](#)
[Windows 2000 Professional](#)
[Windows 2000 Server](#)
[Windows 98/95/CE](#)
[Windows NT](#)
[Windows Web Svcs \(IIS\)](#)

Navigate by Task**IT Solutions****Career and Training****Columns****Downloads****Troubleshoot****TechNet Community****Using TechNet****Developer**[Questions or Comments?](#)

Planning for ASP

John Meade, Web Technology Writer

Internet Information Services Documentation Team

Microsoft Corporation

August 16, 1999

[Send this document to a colleague](#)[Printer-friendly version](#)

Topics on this Page

- ▼ [Introduction](#)
- ▼ [ASP or Browser Scripts?](#)
- ▼ [ASP Administration Planning](#)
- ▼ [Setting Scripting Standards](#)
- ▼ [Conclusion](#)
- ▼ [Resources](#)

Introduction

Active Server Pages (ASP) is a server-side scripting technology that dynamically locates and delivers information requested by users. To be successful with ASP, you should plan the use of good practices in developing and managing your projects. This article is written for systems administrators, Web authors, and Web application developers planning, or considering using, ASP. It presents information that will help plan the development and administration of secure, well organized, easily understood ASP applications.

ASP or Browser Scripts?

Given the requirement for a scripting solution, you will want to determine whether it is better to write ASP pages or write browser-based scripts, and you will want to make this determination early in the development process.

Plan to use ASP when you need to:

- customize Web pages with information gathered from back-end sources and tailored to the needs of the user.
- collect data submitted by site visitors using HTML forms.
- deliver pages that are formatted for particular brands and versions of browsers.

To avoid placing needless burdens on your servers, plan to use browser-based scripting for such tasks as validating user-submitted data, calculations, and the selection of simple conditional output.

This article covers three aspects of ASP application planning:

- preventing common problems
- organizing ASP application directories and files
- setting standards for ASP scripts.

Other aspects of ASP development, such as performance and reliability, will be covered in future articles.

As useful as ASP pages are, it is important to be aware of some of the risks of developing and deploying them. ASP pages that connect users with back-end data resources contain information about how to locate and access enterprise data, and may also contain your enterprise business rules, customer data, and other sensitive information. Consequently, it is important to be aware of the potential for exposing sensitive information to unauthorized people. While developing and deploying ASP it is possible to make mistakes that would allow the wrong people to acquire this information. Furthermore, by failing to use good development and administration practices you could end up with an ever-growing collection of performance and reliability problems instead of a growing number of satisfied users.

ASP Administration Planning

ASP applications are stored in your file system. An ASP application is a collection of ASP pages, along with included HTML pages and components that the application requires. When you define an application, you use Internet Service Manager to designate the application's starting-point directory in your Web site. Every file and folder under the starting-point directory in your Web site is considered part of the application until another starting-point directory is found. Thus, you use directory boundaries to define the scope of an ASP application.

This section provides a template and guidance for directory structure and access permissions for ASP applications. Following this template will help you establish an ASP application storage organization model that will achieve consistency, reliability, and security no matter how large and complex your ASP applications become.

The organization and attributes given to the directories and files in the list below are more important than the names used.

```
/Application_Name
  Default.htm
  Global.asa
  /Classes
  /Content
  /ASP
    *.asp
  /HTM
    *.htm
  /Images
  /Media
  /Themes
```

/Data (not in site directory)
/DLL (not in site directory)
/Helper_Files (not in site directory)

Application Root Directory The application root directory name should clearly convey the theme of the site. For example, an application for financial research might be named /Financial_Research. Avoid application root names that might be misidentified as standard subdirectories of a site, such as /Media or /Content. Also, avoid names that read like part numbers or codes, such as /FR98346A.

To avoid adversely affecting production sites, develop the application in a development test environment. An easy way to do this is to develop new applications under the IIS HTTP root directory, /InetPub/wwwroot, then move them to the same directory under /InetPub/wwwroot in the production environment when they are ready.

Note /InetPub/wwwroot is the home location for all Microsoft® Visual InterDev™ and Microsoft FrontPage™ Web documents. Moving a Web application to a storage location not under /InetPub/wwwroot makes it inaccessible to these tools.

The root directory of every application should contain at least these files:

- Default.htm or Default.asp
- Global.asa

Default.htm or Default.asp should be the default home page for the application, and the server default should be set accordingly using Internet Service Manager. Taking these two steps enables users to find sites in your organization consistently, by typing the server address plus the application root directory name. For example, a user can access MSDN Online by entering msdn.microsoft.com. Entering the name of the home page is not necessary.

The file Global.asa specifies event scripts, declares objects that have application or session scope, and declares type libraries. For example, Global.asa scripts make application- and session-scope variables available at startup. Global.asa must be stored in the application root directory.

/Classes The /Classes directory holds any Java classes used by the application and requires execute permissions.

/Content The /Content directory holds all pages (except Default.htm) and media that may be retrieved directly by a user of the site.

/ASP The /ASP subdirectory of /Content contains all pages with server-side scripting. This directory must contain execute permissions so that ASP can execute the page scripts. Do not assign read permissions to this

directory because .asp pages may contain sensitive information about business rules and access to data resources. Storing all scripted pages here simplifies permissions management and site security.

/HTM The /HTM subdirectory of /Content contains all pages containing only standard HTML. This directory is read-only, and does not have execute permissions. A page containing server-side scripts stored here will not execute.

/Images A subdirectory of /Content, the /Images folder should contain graphics that are used independently of theme-related images, such as standard buttons and icons (see below).

/Media A subdirectory of /Content, /Media should contain subdirectories for audio, images, animation files, .avi files, and similar items used throughout the application.

/Themes A subdirectory of the /Content directory. Use the /Themes directory to enable application-wide changes to the look of a site. The directory should contain style sheets, bullets, buttons, icons, rules, and similar items organized so that you can change the look of an application by changing any or all the theme-related items easily. Each item in the /Theme directory can be linked dynamically by setting an application variable to its virtual path.

/Data This directory should contain all database access information such as SQL scripts, file-based dataset names or other similar data needed by the application. Do not place this directory under the site directory, as this could give an unauthorized user access to business rules and private data.

/DLLs This directory contains COM components and Visual Basic® 6.0 run-time DLLs, such as Vbrun500.dll and Msvbvm50.dll. Do not place this directory under the site directory, as this could give a hacker access to business rules and private data.

/Helper_Files This directory holds server-side includes or text-formatted files that make information available across the application, or many applications. For security reasons, the directory containing helper files should not be stored in the published Web space (the Web site directories identifiable to users).

File-Extension Standards

The file-naming conventions described in this section provide useful standards for reliable, consistent, and secure document pages.

Use of the .asp extension is required for pages that contain server-side scripting. It is a good policy use the .asp extension for pages that are likely to contain scripting in the future, even if they do not initially. To save server resources and to minimize delays when

serving pages, use the .htm extension for files that do not, and will not, require server-side script execution.

For consistency and ease of maintenance, use include files (.inc) to make specific information available to more than one referring page; changes to include files are distributed to all the pages that call them. Use text files (.txt) for plain text-formatted data files to be included in a page.

Do not use .inc for pages containing scripts. If a user manages to display such pages, any business rules in the scripts will be exposed. Use the .asp extension for all pages containing scripting, or for which scripting is planned, to avoid displaying proprietary information coded in ASP scripts.

Setting Scripting Standards

If you have organized, named, and secured your ASP application file spaces you are halfway to a good start with planning and setting up for ASP. You will also want to establish standards for page scripts to make them easy to read by any page author on the team.

Script Styles for Readability

To enhance readability, establish the following script style conventions for ASP pages. The styles apply to scripts written in Microsoft Visual Basic® Scripting Edition (VBScript) or JScript®. As with conventions for directories and files, it is more important to address the issues raised in the discussions that follow than to apply precisely the convention given. When deciding how to define your scripting conventions, allow for the fact that most ASP pages contain some plain HTML as well.

This section will get you started establishing scripting standards. A more extensive script style guide reference will be published on this site in the near future.

Comments in Scripts Comments should help any script author looking at code begin to understand it immediately. In addition, comments should explain the intent of the code or summarize what the code does, not simply repeat what the code says.

Write consistent comment blocks near the top of each page listing the file name, the group developing the file (not the individual; e-mail should go to a group alias), the date the file was developed, the HTML and scripting standards followed, and dated descriptions of all changes made.

Use comments to explain obscure or complex code, that is, any coding that would take a script author more than a few seconds to decipher. Do not leave a phrase such as the following without a comment:

```
If Err = LOCK Then
```

Scripts that are commented out should be deleted unless

they are placeholders, in which case they should be labeled as such.

Insert each comment with its corresponding code. Inline comments should appear two spaces after the corresponding code. Comments beginning on a new line should be set off with a blank line.

Example:

```
<%
Dim intVariable 'Explicitly declare variable.
'Assign the variable an integer value.
intVariable = 5
%>
```

If a single comment spans multiple lines, each line must begin with the standard VBScript comment symbol ('). Large, multistatement comment blocks should be formatted as in the example below.

Example:

```
Sub ShowIt()
'=====
'This procedure is called when the
'user selects a language.
'
'It displays an appropriate select
'item based on their language choice.
'
'The method choices are each contained
'in a separate div.
'=====

Dim vntCurrLang
vntCurrLang = document.all.langselect.value
Select Case vntCurrLang
Case "C"
document.all.cdiv.style.display = ""
Case "VB"
document.all.vbdiv.style.display = ""
Case "J"
document.all.javadiv.style.display = ""
End Select
End Sub
```

Multiline comments in JScript begin with /* and end with */. Large, multistatement comment blocks should be formatted as in the example below.

Example:

```
function showIt()

/*****
** This is a large comment block.
**
** This procedure is called when the user
** selects a language.
**
** It displays an appropriate select item
** based on their language choice.
**
** The method choices are each contained
** in a separate div.
*****/

{
var vntCurrLang = document.all
...
}
```

Constant Names Use all uppercase when naming

constants to distinguish them from other elements. An underscore can be used to separate terms when necessary.

Example:

```
Const MIN_QUAL = 25
```

Context Switching For readability, try to minimize switching between HTML and scripts. When possible, use a few large blocks of script on a page instead of a larger number of scattered fragments.

Indentation Indentation makes the logical structure of the code more clear.

Place a script consisting of more than one line on a line below the script delimiter, blocking and indenting it two spaces. Place a single-line script on the same line as the delimiter.

Indent everything between ASP delimiters (`<% U %>`) at least two spaces. Also, add two spaces of indentation for each:

- Break in logic.
- Nested statement or HTML element.
- Body of a function.
- Body of a loop from its controlling code.

The following examples illustrate some of the indentation rules for scripts written in either VBScript or JScript.

Single-line Script

```
<% Dim strLastName %>
```

Script with Nested Logic:

```
<%  
'This example demonstrates a script with  
'a nested block of logic.  
  
Dim vntOutput  
Set vntExample = Server.CreateObject  
("MyComponents.Component.1")  
vntOutput = vntExample.Text  
  
If vntOutput = "" Then  
Response.Write "An error has occurred"  
Else  
Response.Write vntOutput  
End If  
>%
```

Scripting a function in VBScript:

```
<%  
Function CalcMortgageRate()  
Statement_1  
End Function  
>%
```

Scripting a function in JScript:

```
<%  
//This is an example of a function.  
function calcMortgageRate()  
{  
statement1  
statement2  
}  
%>
```

Conclusion

If you establish and maintain good directory structure and permissions, and effective page script standards, you will find that your ASP applications are more manageable, more secure, and easily read and understood by systems administrators and development teams.

Resources

Books

Internet Information Server Resource Kit, 1998,
Redmond: Microsoft Press
<http://mspress.microsoft.com/books/1398.htm>

This book covers all aspects of using IIS 4.0, including development of Web applications, and contains an appendix on ASP standards.

Web Links

<http://msdn.microsoft.com>

The Microsoft Developer Network (MSDN) is the comprehensive resource for all Microsoft development technologies, including ASP.

<http://www.windowstechedge.com/wte/wte-1999-01/wte-01-asp.html>

This Window TechEdge article by Brooks Talley instructs you to plan your ASP application development as if expansion and complication are inevitable. Talley emphasizes standardization and reuse as keys to effective ASP project planning.

Last updated November 12, 2000

© 1999 Microsoft Corporation. All rights reserved. Terms of Use.



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#)
Navigate**Index**Top IT Tasks 

Search TechNet

Navigate by Product

[Application Center](#)
[BizTalk Server](#)
[Commerce Server](#)
[Exchange Server](#)
[Host Integration Server](#)
[Internet Security & Acceleration Server](#)
[Office](#)
[Site Server Commerce](#)
[Small Business Server](#)
[SQL Server](#)
[Systems Management Server](#)
[Visio](#)
[Windows 2000 Professional](#)
[Windows 2000 Server](#)
[Windows 98/95/CE](#)
[Windows NT](#)
Windows Web Svcs (IIS)

Navigate by Task**IT Solutions****Career and Training****Columns****Downloads****Troubleshoot****TechNet Community****Using TechNet****Developer**

Questions or Comments?

Chapter 2 - Understanding the Internet and Internet Information Server

Prior to planning and implementing your Microsoft Internet Information Server site should understand each of the components involved in establishing an Internet Information Server site.

This chapter answers the following questions:

- What is the Internet?
- What is an intranet?
- What is Internet Explorer?
- What is Internet Information Server?

What is the Internet?

The Internet is a global network of computers that communicate using a common language. It is similar to the international telephone system— no one owns or controls the whole thing, but it is connected in a way that makes it work like one big network.

The World Wide Web (WWW) gives you a graphical, easy-to-navigate interface for documents on the Internet. These documents, as well as the links between them, comprise a "web" of information.

Files, or "pages," on the Web are interconnected. You connect to other pages by clicking special text or graphics, which are called hyperlinks.

Pages can contain news, images, movies, sounds— just about anything. These pages can be located on computers anywhere in the world. When you are connected to the Web, you have equal access to information worldwide; there are no additional long-distance charges or restrictions.

Hyperlinks are underlined or bordered words and graphics that have Web addresses embedded in them. By clicking a hyperlink, you jump to a particular page in a particular Web site. You can easily identify a hyperlink. Hyperlink text is a different color from the rest of the text on a Web page.

Each Web page, including a Web site's home page, has a unique address called a Uniform Resource Locator (URL), for example, <http://www.microsoft.com/home.htm>. Domain System (DNS) names are used on the Internet.

What is an Intranet?

In this book, "intranet" refers to any TCP/IP network that is not connected to the Internet. An Internet Information Server can be configured to provide your intranet with the same features and services found on the Internet, such as hypertext pages (which can contain text, hyperlinks, images, and sounds), client/server applications, and database access.

What is Internet Explorer?

Microsoft Internet Explorer is a Web browser. Just as Microsoft® Word is a tool to processing, or Microsoft Excel is a tool to do spreadsheets and calculations, Internet Explorer is a browser, or a tool for navigating and accessing information on the Web.

The toolbar provides a range of detailed functions and commands for managing the browser. The address bar below the toolbar displays the current Web site address accessed. To go to a new Web site, you type the site's URL directly into the white space of this bar. When you have finished typing, press ENTER on your keyboard.

Microsoft Internet Information Server includes a version of Internet Explorer for each Windows version:

- Internet Explorer for Windows NT
- Internet Explorer for Windows® for Workgroups and Windows version 3.1
- Internet Explorer for Windows 95

What is Internet Information Server?

Microsoft Internet Information Server is a network file and application server that transmits information by using the HyperText Transport Protocol (HTTP).

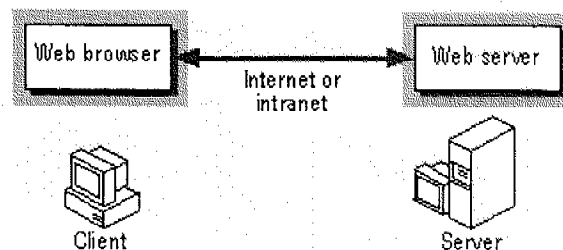
What Can I Do with Internet Information Server?

The creative possibilities of what you can offer on a Microsoft Internet Information Server are endless. Some familiar uses are to:

- Publish a "home page" on the Internet for your business featuring a newsletter, sales information, or employment opportunities.
- Publish a catalog and take orders from customers.
- Publish interactive programs.
- Provide your remote sales force easy access to your sales database.
- Use an order-tracking database.

How Does Internet Information Server Work?

The Web is fundamentally a system of requests and responses. The Microsoft Internet Information Server responds to Web browser requests for information. The Internet Information Server listens for requests from users on the network using the WWW.



Browser Requests

Browser request syntax determines what the server will do with the request. Requests are in the form of an URL.

URL Syntax

URL syntax is a specific sequence of protocol, domain name, and path to the requested resource.

information, as described in the table below. Protocol is the application used to gain information; for example, HyperText Transport Protocol (HTTP). Domain name is the name registered in DNS. The path is the path on the server to the requested information.

Protocol	Domain Name	Path to Information
http://	www.microsoft.com	/backoffice
https:// (secure HTTP)	www.company.com	/catalog/orders.htm
gopher://	gopher.college.edu	/research/astronomy/index.htm
ftp://	orion.bureau.gov	/stars/alpha quadrant/starlist.txt

Request Syntax

Just as Microsoft Word documents use the convention *Filename.doc*, and programs use the convention *Programname.exe*, the path to information determines whether the request is for a static HyperText Markup Language (HTML) page, for a dynamic HTML page, or for a directory listing. Sometimes the path includes parameters, or data the Information Server will process before returning a dynamic page.

In all cases the server replies with an HTML page (or an error message). Example request types are listed in the following table:

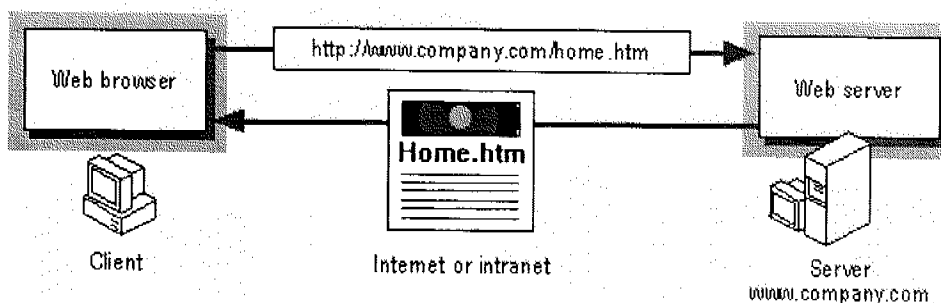
Request Type	URL
Static HTML page	http://www.microsoft.com/backoffice/home.htm
ISAPI application	http://www.msn.com/custom/page1.dll?CUST=on
Internet Database Connector	http://www.microsoft.com/feedback/input.idc
CGI script	http://www.company.com/calculator/add.pl?2.2
Directory listing	ftp://orion.nasa.gov/stars/alpha quadrant/list

Information Server Response

Responses are typically in the form of an HTML page. The returned page can be one of three types: a static HTML page, a dynamic HTML page, or a directory listing page.

Static Pages

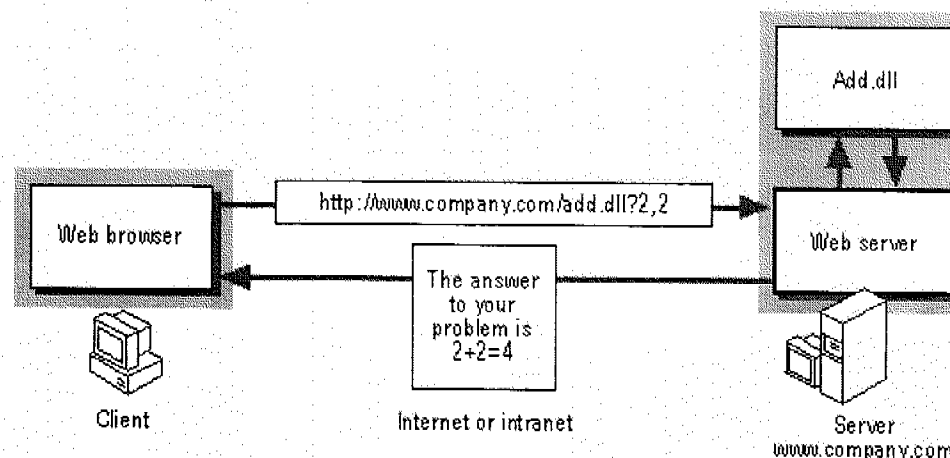
Static pages are static HTML pages that are prepared in advance of the request. The Information Server returns the HTML pages to the user, but takes no special action. A user requests a static page by typing in an URL (in the following illustration, <http://www.company.com/home.htm>) or by clicking a link pointing to an URL. The request is sent to the server. The server responds by returning the static HTML page.



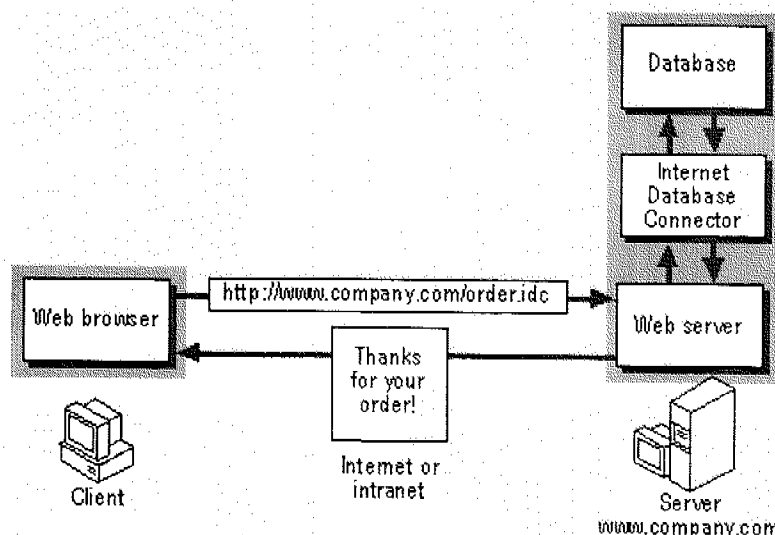
Dynamic Pages

Dynamic pages are created in response to a user's request. The user requests a dynamic page by clicking a link pointing to an URL, or by clicking a button on a form, which sends the data in the form to the server. The server uses any data supplied by the user to run the specified script or application or to query or post data to a database. The server then returns the results to the user in an HTML page.

The following illustration shows how a user can send a query to an Internet Server (ISAPI) application that adds two numbers. The user types the two numbers to be added, then clicks a button, which in turn sends the two numbers to the server. The server processes the numbers, then returns the results to the user in an HTML page.

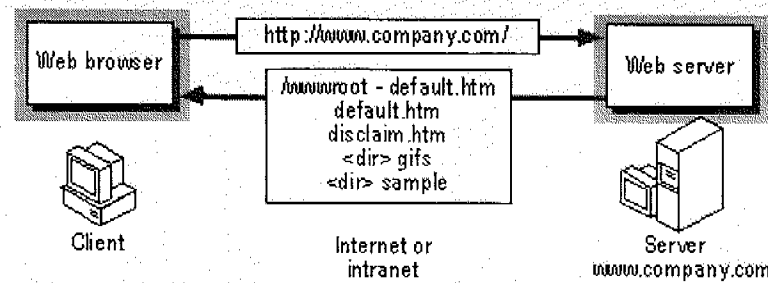


The following illustration shows a user posting an order to a database using the Internet Database Connector. The user completes a form, then clicks a button, which in turn sends the data in the form to the server. The server posts the data to a database, then returns the order by sending an HTML page.



Directory Listings

If users might send queries without specifying a particular file, you might want to configure your server for directory browsing. If directory browsing is configured, a directory listing (a hypertext version of a File Manager listing) is returned to the user in the form of a page. The user can then jump to the appropriate file by clicking it in the directory listing.



In summary, Internet Information Server responds to user requests with an HTML page. This page can be a static page that is already prepared, or it can be generated in response to information that the user provides, or it can be a directory listing that is created automatically from a listing of the available files and directories.

How Do I Use Internet Information Server?

Internet Information Server is flexible enough to perform many important functions for your organization. It is scalable from supporting a single-server site to supporting multiple-server installations. For example, www.microsoft.com and www.msn.com are the busiest Web sites on the Internet today, and both use multiple servers running Microsoft Internet Information Server.

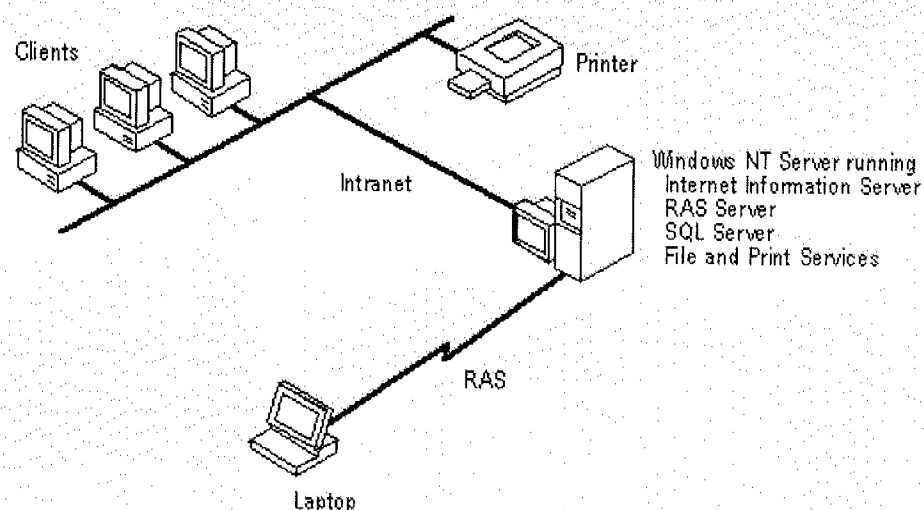
One of the primary factors that determines the configuration and use of Internet Information Server is whether it will be used internally by employees on your intranet or whether it will be connected to the Internet.

The following scenarios are intended to help you understand the range of possibilities using Internet Information Server.

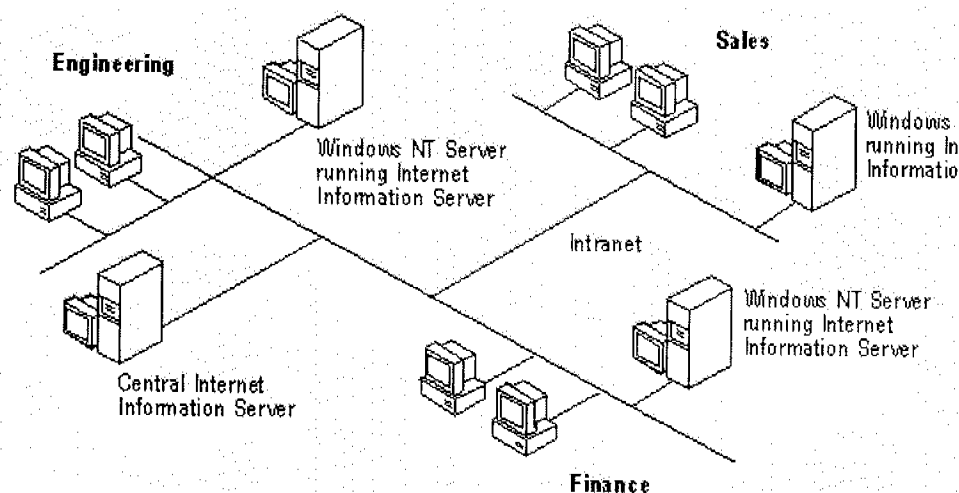
Intranet Scenarios

Internet Information Server integrates well into almost any existing environment. Internet Information Server integrates Windows NT security and networking, so you can add the software to an existing computer and use existing user accounts. It is not necessary to use a dedicated computer to run Internet Information Server.

For example, in a small workgroup you can add Internet Information Server to an existing file and print server. The WWW server can host personal Webstyle pages, custom workgroup applications, serve as an interface to the workgroup's Sequential Query Language (SQL) database, or use Remote Access Service (RAS) to provide dial-up access to the workgroup's resources from remote sites.

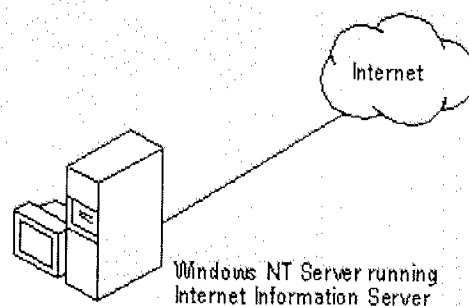


In a larger business with multiple departments or workgroups, each department might have its own Internet Information Server on an existing file server for workgroup-specific information. A central information server might be used for company-wide information, such as an employee manual or company directory.



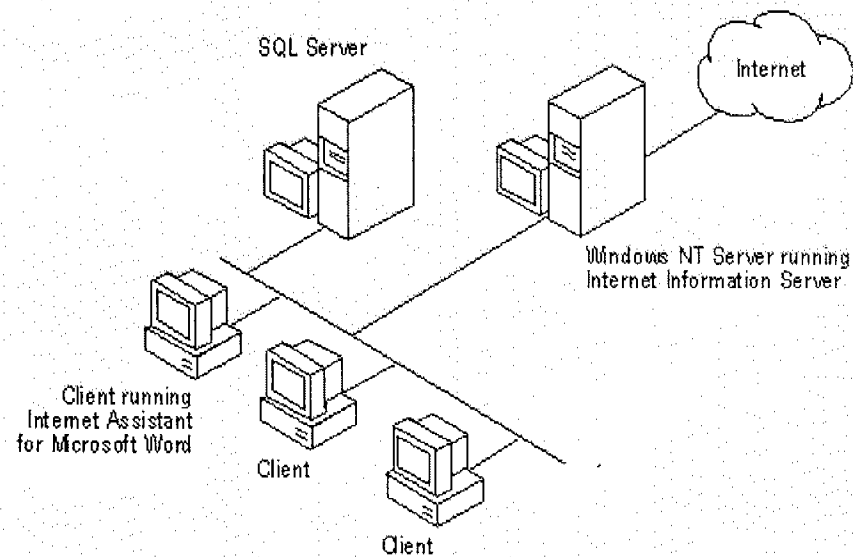
Internet Scenarios

Internet Information Server can function as a simple dedicated WWW server on the Internet, as shown in the following illustration.

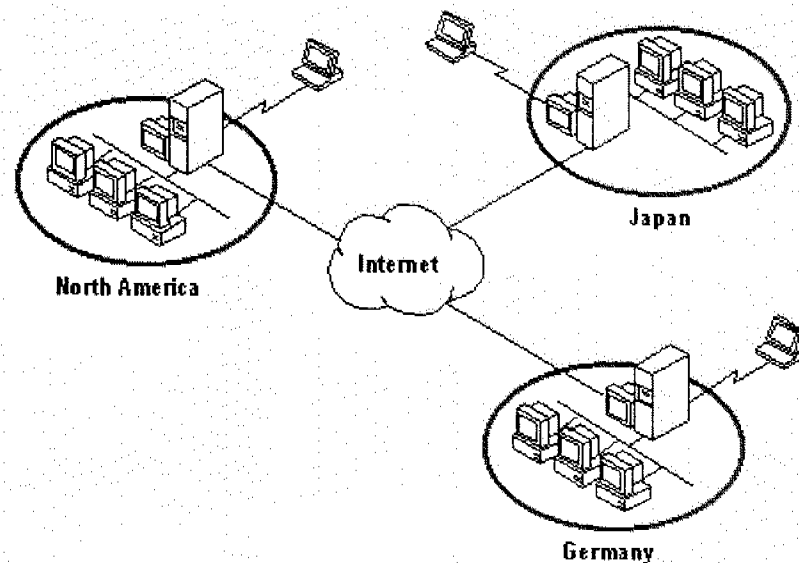


In larger sites you can provide access from your internal network to the Internet Information Server, allowing employees to browse the server or to use authoring

such as Internet Assistant for Microsoft Word, to create content for your server.



Internet Information Server's integration with all of the Windows NT services can create servers with multiple functions. For example, a company with sites in different parts of the world can use Internet Information Server to provide communication between sites with the added flexibility of Internet access. You can even add RAS to an Internet Information Server to provide dial-up access to your intranet or the Internet.



Note Many scenarios for connecting to the Internet involve third-party routers or devices that filter network packets between your computer and the Internet. Route and other security devices are not indicated in the preceding illustrations.

Last updated January 12, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#)
Navigate**Index****Top IT Tasks** [Select from this list](#)**Search TechNet****▣ Navigate by Product**

[Application Center](#)
[BizTalk Server](#)
[Commerce Server](#)
[Exchange Server](#)
[Host Integration Server](#)
[Internet Security & Acceleration Server](#)
[Office](#)
[Site Server Commerce](#)
[Small Business Server](#)
[SQL Server](#)
[Systems Management Server](#)
[Visio](#)
[Windows 2000 Professional](#)
[Windows 2000 Server](#)
[Windows 98/95/CE](#)
[Windows NT](#)
Windows Web Svcs (IIS)

▣ Navigate by Task**▣ IT Solutions****▣ Career and Training****▣ Columns****▣ Downloads****▣ Troubleshoot****▣ TechNet Community****▣ Using TechNet****▪ Developer**[Questions or Comments?](#)**Database Connectivity****Topics on this Page**

- ▼ [The Dao of Database](#)
- ▼ [Database Products](#)
- ▼ [Internet Databases](#)
- ▼ [Data Access Components](#)
- ▼ [Microsoft Transaction Server](#)
- ▼ [A Word About Database Security](#)
- ▼ [Summary](#)
- ▼ [Review Questions](#)

Matthew Strebe and Charles Perkins

Chapter 13 from *MCSE: Internet Information Server 4 Study Guide*, published by Sybex, Inc.

Nearly all commercial transactions are processed by databases; databases are the software engines of our economy. The business use of the Internet, for more than ad brochure Web pages and e-mail, depends on the ability to interface Web sites with databases. Most important commercial information is stored in databases, and the ability to interface with these databases over the Internet allows consumers to find and purchase the products they need without involving a human operator or making a phone call.

Databases are also useful for reporting tables of information, such as portions of phone books or catalogs of product information. Databases are appropriate for any application that requires the storage or retrieval of data.

This chapter covers the database connectivity options that IIS and the Microsoft database products Access and SQL Server support. IIS supports any standard database solution, but Access support for Internet integration makes publishing databases on the Web especially easy.

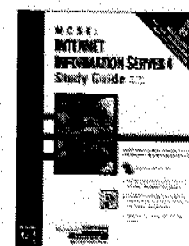
This chapter covers the following topics:

- How databases work
- The differences between databases
- The criteria you should use to select a database product
- How IIS supports the connection of databases to the Internet

You will also learn how to create Web pages to support database connectivity, using a sample Contacts database.

Send this document to a colleague

Printer-friendly version

[Click to order](#)

The Dao of Database

If you work with computers, you've probably heard a lot about databases. Often they are presented as mysterious solutions to problems as widely disparate as manufacturing control, workflow, order entry, and document management. You've probably heard many different software systems described as databases (like the DNS system presented in Chapter 5). Trying to find a common thread among these different uses can be a bit confusing.

Databases are nothing more than organized collections of data. Your telephone book is a perfect example of a manual database, one that contains four elements of data:

- Last Name or Business Name
- First Name
- Address (optional)
- Telephone number

In a database, these elements are called *fields* (or *columns* because when laid out as a table, the elements form columns). Entries in the phone book (e.g., Valentino, Rudolph, 555-1212) are called *records* (or *rows* because when laid out as a table, they form rows). If we printed a portion of a telephone database, it might look something like Table 13.1.

Table 13.1 A fictitious Phone Book

Entry	Last Name	First Name	Address	Telephone Number
1	Foot	B.	13 Forest Way	555-9192
2	Kringle	Kristopher	34th St.	555-4242
3	Kruger	Frederick	683 Elm St.	555-1234
4	Valentino	Rudolph	3434 Hollywood Blvd.	555-3452
5	West	Mae	789 Vine St.	555-6547

Table 13.1 represents a *table*. A table contains lists of information of the same structure. A *relational database* comprises one or more related tables. Relational databases keep track of how a record in one table (say, a list of customers) relates to records in another table (say, orders).

Suppose you order books from a company on the Web. If you had to reenter your name and address every time you ordered a book, you'd quickly tire of their system. Relational databases allow you to enter your address information once into an address table. Later, when you

place an order, the company can simply look up your address from the address table using some key information like your name or a customer ID issued when you started using the service. A *key* is a field that is stored in both tables that relates (or *joins*) the information in the two tables. When you place an order, your customer ID is stored in the order table. When it comes time to print the shipping label, the database *queries* (or asks for the record or records that satisfy the search criteria) the address table by your customer ID. Your address is the result of the query, which is printed on the shipping label. Because you can have only one address but can place many orders, this type of data relationship is called a *one-to-many* relationship.

Now let's say the company wanted to offer a discount to its best customers. A relational database allows the merchant to query the orders table by customer ID. This query will return all the orders you've made. If the number of orders you've made satisfies the cutoff for the discount program, you'll automatically receive a discount on your next order.

After you enter your order, the database will automatically run your credit card for the required amount and print an order to ship the book (with a mailing label) in the shipping department. The only human assistance required is from the person who looks at the order printouts, retrieves the specific book from the warehouse, and sticks the mailing label on it—and even these steps are being automated!

Database Products

In the early days of database management (before 1990), two distinct types of database management applications existed:

Relational client/server databases (structured query language (SQL) databases such as Oracle, DB2, or Informix) tracked millions of records and supported thousands of simultaneous users. The software was expensive; the hardware even more so.

Flat-file PC LAN databases (dBase, FoxPro, Pick, or Clarion) were based on proprietary scripting languages and did not support real data relationships. These databases were intended for small workgroups or businesses with limited data processing needs. The software was cheap and ran on PCs.

These two database camps had very little to do with one another. Client/server relational databases were the acknowledged solution for medium to large databases that supported large numbers of simultaneous users, and PC LAN databases were the acknowledged solution for single computer or workgroup databases for small businesses. Entirely different groups developed software solutions for each, and very little cross-development occurred. The programming and information management skills for the two types were very different, so there wasn't much point

in trying to make them work together. PCs simply didn't have the computing power available to resolve relational queries or to manage large amounts of information.

That situation has, of course, changed. PCs are now as powerful as the minicomputers of ten years ago; they can run fully relational database management software and can act as the client, the server, or both in the distributed database model. Nearly all current PC flat-file database software packages support network interoperability with SQL servers, and most of them are either truly relational or will automatically generate code that makes them work like relational databases. In addition, SQL servers can now be run on standard PC servers to support thousands of simultaneous users.

This merging of technology can make choosing a database somewhat confusing.

LAN Databases

Low-end, LAN-based nonrelational (or *flat file*) databases are stored on file servers (not application servers) or on PCs. The database and the application are stored together in the same directory. Data is retrieved programmatically through written scripts, rather than automatically with defined relationships. If more than one client uses the same database at the same time, simple record locks prevent data corruption.

Microsoft Access

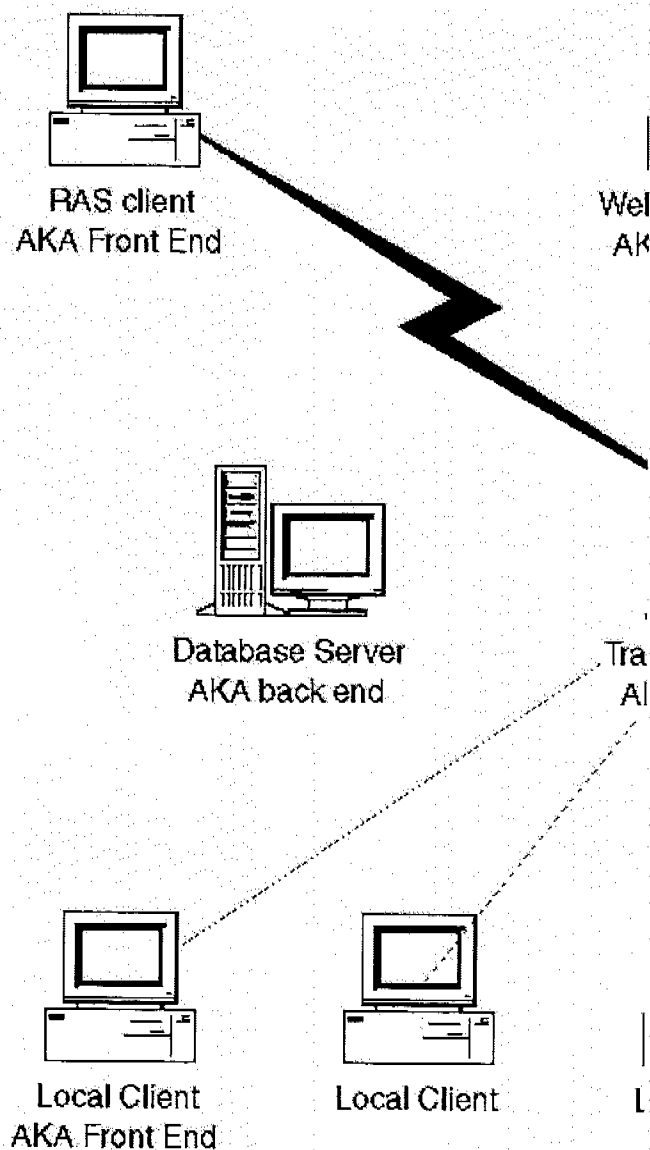
Microsoft Access has blurred the line between application servers and LAN-based application packages. Although designed to run on the user's workstation rather than a server computer, Access is truly relational and supports SQL queries. Access 97 supports a distinct split between the client application and the server-based data (several Access programs on different client computers can simultaneously access the same database files on the file server). However, its Jet database engine supports neither as many simultaneous client connections nor the robust fault-tolerance mechanisms that true client/server database servers support.

Access Licensing Issues Microsoft Access is licensed per copy of Access in use. For instance, if you have five people using Microsoft Access at the same time, you must have five licenses to use Microsoft Access. However, you do not have to pay for concurrent use of an Access database. If you publish your Access database using the Publish to the Internet wizard, you don't have to pay a licensing fee or a client access license for computers that attach to an Access database without using Access, such as those coming in over the Internet.

Consequently, you can create your database with a single copy of Access, publish it to your Internet server, and have many people legally access it via their Web browsers. The Jet database engine (upon which Access is built) does have a hard limit of 64 concurrent accesses to the same database, but it will probably not work very well past 25 concurrent users as it is designed for lighter workloads.

Client/Server Databases

High-end, client/server relational databases run on database application servers and serve queries to thousands of clients simultaneously. The database files are stored on the server, and clients send query requests using the *structured query language* (SQL) to the server. The server responds with the record or records that fulfill the query, which are then edited or processed on the client. Client software packages use the data stored on the database server. These client packages are usually custom written in Visual C++, Visual Basic, Access, or Delphi (Borland's rapid application design tool based on an object-oriented version of the Pascal programming language). Figure 13.1 shows a graphical representation of a typical client server database.



If your browser does not support inline frames, click [here](#) to view on a separate page.

Figure 13.1 A Database diagram showing various

components

SQL Server

SQL Server is Microsoft's database application server software package. SQL Server runs on a server (generally a server of its own) and responds to query requests for the data it stores from clients. SQL Server is a strong contender in the SQL server market, but it runs only on Windows NT. (Most other SQL server packages run on VMS, UNIX, or other traditional mainframe and minicomputer operating systems.)

SQL Server does not come with client access software beyond a few tools to test your database. You must create your own client software to access a SQL server. Perhaps the fastest way to create a client for SQL Server is to use Microsoft Access with linked tables to a data source provided by a SQL server. By using Access as a database front end, you can then use the Publish to the Internet wizard built into Access to create Active Server Pages for IIS. Web browsers will then be able to connect directly to your SQL Web server through Web pages stored on your Web server. (Access is used only to create the Active Server Pages automatically, so it's out of the process at this point.) This approach means that you won't have to pay much for custom client software development and that your client interface can access the SQL server from anywhere on the planet.

The SQL Server Internet License SQL Server requires a client access license for every concurrent user that attaches to the SQL data source by any means. Therefore, if 1,000 people are accessing your SQL server over the Internet and 200 people are accessing it directly via your in-house LAN, you will need 1,200 client access licenses. At current prices, your cost would be about \$6,000.

More importantly, you may not even know how many people will access your site via the Internet, so you won't really know how many licenses to buy. Microsoft provides a solution: For a reasonable sum (currently about \$4,000), you can get a SQL Server license that allows unlimited use via the Internet. Once you have this license, you won't need to worry about the number of simultaneous Internet users on your site.

The unlimited Internet access license does not clearly state if it covers intranet use, nor is it clear on whether the license covers client software on your local network. You should check with Microsoft regarding these licensing issues if you intend to use the SQL Server Internet license on your intranet.

You do not have to purchase the Internet unlimited access license if you don't mind limiting your SQL server to a smaller number of concurrent users. If you intend to provide service to fewer than about 100 users, it makes more sense to simply purchase the number of licenses you need and limit the server to that number. You can add licenses as you please, and Microsoft will give a credit toward an unlimited license based on the number you've already purchased. Contact a Microsoft sales

representative to determine exactly what level of licensing is most cost-effective for your application.

Internet Databases

Publishing data on the Internet can be easy or difficult depending on the tools you use and your needs. If you simply want to report data or produce a static catalog or product listing, you can generate a report in HTML format (or even text) and store it on your Web server. If you need current information every time a database is accessed, you can create query files that generate reports automatically from your database and convert the reports to HTML output files using CGI scripts or the Internet Database Connector for your database product. Finally, if you want to interact with your client/server database over the Internet as a full-fledged database client, you'll need to create a sophisticated set of Web pages.

Microsoft Exam Objective Configure IIS to connect to a database. Tasks include:

- Configuring ODBC

Three styles of database publications appear on the Internet:

Static Static HTML databases are unchanging HTML pages that are produced as output reports from a live database manually. Static HTML databases do not reflect changes to the database since the last production of the page, and they do not put much processing load on the Internet server. Static pages are useful for catalogs that don't change much, are heavily accessed, or are located on servers that you do not own (because they don't require execute permissions).

Dynamic Dynamic HTML databases are output-only databases that are produced automatically each time the page is accessed, providing complete snapshots of the database at any time. Producing the HTML output puts some processing load on the Web server and requires the Web server to run software to interface to the database.

Interactive Interactive HTML databases are input/output databases that are controlled through ISAPI applications, CGI scripts, or Active Server Pages. Interactive HTML databases can be viewed and updated live over the Internet. Interactive HTML databases put a serious load on the Internet server, which should actually be configured as a database server because of the traffic load involved and because it will have to run or connect to the database server software.

Each level of interactivity is appropriate in different circumstances and for different types of data. Some data doesn't change often, so it doesn't make sense to go to a lot of trouble to make an interactive HTML database site to

support it. Other information (like stock quotes) is worthless if it isn't current, so you really need to make it dynamic if you publish it at all. If you are looking at the Internet as an inexpensive way to extend your company's wide area networked database, you will require true interactivity.

A Database Example

The companion CD-ROM contains a sample Access database that stores addresses and phone numbers. This type of database is easy to create with a tool like Access, and many organizations use similar databases to maintain a unified company directory of contacts.

Maintaining a unified directory between branch offices, however, rarely makes much sense because the difficulty involved in setting up a wide area network and creating a complex self-replicating database between the sites isn't justified for this simple database. This level of effort is usually worthwhile only for order-entry databases or other mission-critical database applications.

Access includes convenient Publish to the Web wizards with which you can create a true client/server database application that uses Access's Jet database engine, IIS as the server, and a Web browser as the client. This easy method makes providing even the simplest database worthwhile—it's so easy that you'll actually spend more time tweaking the results to look good than you will generating the pages.

Combined with Active Server Pages technology to create the views, publishing databases on the Web can be as interactive as working with them locally. Earlier versions of Access could create static HTML pages of information, useful for publishing catalogs and phone lists on the Web, but you really couldn't add new data to the site without a lot of work. ActiveX and Active Server Pages make truly interacting with a database easy.

Figure 13.2 shows the form used to add or edit a phone book entry in Access 97, the application used to create the database.

Addresses

Address

Honorific	FirstName	MiddleName	LastName
Ms.	Mae		West

Street
789 Vine St.

City	State	Country	Zip
Hollywood	CA	US	99321-

Phone	Fax
(310) 555-6547	(310) 555-6547

Record: 1 2 3 4 5 6 7 8 9 10 of

Figure 13.2 An Internet view of an Access database

Compare this form to Figure 13.3, which shows Internet Explorer browsing an Active Server Page Web site that was automatically created by Access 97 from the same database.

Addresses - Internet Explorer

File Edit View Go Favorites Help

Address

Honorific	FirstName	MiddleName	LastName
Ms.	Mae		West

Street
789 Vine St.

City	State	Country	Zip
Hollywood	CA	US	99321-

Phone	Fax
3105556547	3105556547

k < > >I >* Commit Delete

If your browser does not support inline frames, click here to view on a separate page.

Figure 13.3 An Access database form

The two are obviously different, yet they share the same structure and feel. Anyone familiar with the first form will have no problem adapting to the Internet-based form. The phone book database is a simple example—any complex database can be published as a complex Web site. You may find you'll have to correct some of the automatically generated Web pages, and some more complex Access controls do not work the same way when published to the Web. More information about these issues is available in the Access online help.

The exercises in the rest of this chapter use the phone book database to introduce the steps you need to take to publish an existing database on the Internet. In the following examples we show you how to create interactive Internet databases. Static and dynamic Internet databases are easier to produce, so you'll be able to figure them out on your own after completing these exercises.

Exercise 13.1 sets up the sample phone book database for publication on the Internet. You will copy the sample database to a folder on your computer, and then create another folder in your wwwroot to hold the Active Server Pages automatically generated by the Access Publish to the Internet wizard in Exercise 13.2.

EXERCISE 13.1**Preparing the Phone Book Database for Publication**

1. Insert the companion CD-ROM in your CD-ROM drive.
2. Double-click My Computer.
3. Double-click your CD-ROM drive.
4. Double-click the Samples folder. Close any other windows that may be open.
5. Double-click My Computer again.
6. Drag the My Computer window to another portion of the screen so that it does not obscure the Samples folder window.
7. Double-click your C:\ drive.
8. Right-click the white background and select New > Folder.
9. Change the name of the folder to phonebook.
10. Drag the phonebook.mdb file from the Samples window to the phonebook folder in the C:\ drive window.
11. Double-click My computer.
12. Double-click your C:\ drive or the drive containing your http root directory.
13. Browse to \InetPub\wwwroot or the http root on your computer.
14. Right-click on the background of the wwwroot window.
15. Select New > Folder.
16. Change the name of the new folder to phonebook.

Database Publication with IIS

Because IIS controls the services that actually publish your database on the Internet, you will have to tell IIS

how you want to deal with the data. You should create a folder to contain the site, create a virtual directory or virtual server to support it, and set the access permissions as appropriate. Remember that Active Server Page scripts must have Execute permission assigned in IIS to function properly. Exercise 13.2 shows you how to set up the Internet Service Manager to serve the sample database.

EXERCISE 13.2

Setting Up the Phone Book Directory with the Internet Service Manager

1. Select Start > Programs > Windows NT 4 Option Pack > Microsoft Internet Information Server > Internet Service Manager.
2. Expand Internet Information Server and your computer name such that the Default Web Site is visible in the scope pane of the management console.
3. Right click Default Web Site and Select New > Virtual Directory.
4. Type **phonebook** in the Alias input line.
5. Click Next.
6. Click Browse.
7. Browse to C:\InetPub\wwwroot\phonebook or its equivalent on your system.
8. Click OK, then click Next.
9. Check the Execute permission in the Access control group.
10. Click Finish.

Open Database Connectivity

The Open Database Connectivity (ODBC) interface is the standard database interface that allows databases to access or link to other software products. ODBC provides a standard method to configure a database as a *data source*, or a publisher of database information. Other software products (like IIS) can then link to those data sources to request (query) information or store information to the database. Most database products that run under Windows support ODBC, so they can function as database engines for IIS. Exercise 13.3 shows you how to configure the Access ODBC driver to publish an Access database.

EXERCISE 13.3

Defining an ODBC Data Source

This exercise requires a complete installation of Microsoft Access 97.

1. Select Start > Settings > Control Panel.
2. Double-click the ODBC Control Panel.
3. Click System DSN.
4. Click Add.
5. Double-click the Microsoft Access Driver. If the Microsoft Access Driver does not appear, reinstall Microsoft Access 97 with the complete installation option.
6. Type **phonebook** in the Data Source Name input

- line.
7. Type **A Web-enabled access phone book** in the Description input line.
8. Click Select to select a database.
9. Browse to the C:\phonebook\phonebook.mdb database and click OK.
10. Click OK.
11. Click OK.
12. Close the control panel window.

Internet Database Connector

The Internet Database Connector (IDC) is an add-on to IIS that allows the dynamic publication of databases. Dynamic HTML databases are databases that are updated every time a Web browser requests them, so they contain up-to-the minute snapshots of the database in question. This method of publication is different from simple static HTML databases, which are published manually whenever the database administrator gets around to it and do not contain changes more recent than the last publication.

IDC works as a plug-in to IIS (actually, it's an ISAPI application). When a browser requests a dynamic database, IIS reads the IDC file and performs the database connection and query instructions contained within it to create an HTML page. This HTML page contains the results of the database query stored in the IDC file and is formatted according to HTML format instructions contained in an associated .htx file. Because the database is required and a new HTML page is created each time a connection is made, data on the Web site is always in sync with data in the actual database.

Tip Remember, the .idc file contains the query for retrieving the database information from the database server. The HTX file contains the instructions for formatting the data as HTML.

Some database products contain wizards to automatically create the .idc and .htx files required by the IDC. When you select the Dynamic HTML option in Microsoft's Publish to the Internet wizard in Access, you are creating the .idc and .htx files required by the IDC.

The IDC has no provisions for accepting database input from the Web, however—it is a one-way publication medium. You will have to use CGI scripts, ISAPI applications, or Active Server Pages if you want to update a database over the Internet.

Before Active Server Pages existed, IDC was created to allow dynamic publication to the Web. Because Active Server Pages supersedes the functionality of IDC and allows far more query and format options, you should use it for any new custom development that is more extensive than simply publishing data with Access. If you want more information on the format of .idc and .htx files, consult the help files that come with IIS or perform a search for documentation on the Microsoft Web site.

Publishing Access Data with Active Server Pages

Dynamic HTML databases are convenient for their purpose, but they still don't allow remote users to add or edit the information they provide. Interactive HTML databases provide this functionality by running code on the Web server to extract information from the remote Web browser and push it into the ODBC-linked database driver running on the Web server. This ODBC driver may simply provide an interface to an existing SQL server, or it may start a database engine to store data into a database file. In either case, the key is the software running on the Web server retrieves the existing client data and stores revised records into the database.

This process used to require quite a bit of trick programming with complex CGI scripts or the use of ISAPI applications dedicated to the task. With Active Server Pages and Access 97's Publish to the Internet wizard, the process of publishing an interactive HTML database is easy. Exercise 13.4 walks you through using Access to publish a sample database on the Internet. This exercise creates the Active Server Web pages that provide access to your ODBC data source. This data source could just as easily be an SQL server, a FoxPro database, or even an Excel spreadsheet—although those products don't automatically produce the Active Server Page code to publish the data.

Warning The directory containing your .asp files must have Execute permission set in the Internet Service Manager, or browsers will simply see your ASP code.

EXERCISE 13.4

Creating an Internet-Enabled Access Database

1. Double click My computer and browse to the C:\Inetpub\wwwroot\phonebook folder or its equivalent on your computer.
2. Double click the phonebook.mdb file. If MS Access does not launch, it is not correctly installed on your system.
3. Select File > Save As HTML.
4. Click Next.
5. Click the All Objects tab.
6. Click Select All.
7. Click Next.
8. Click Browse.
9. Double-click Default.htm and then click Next.
10. Click Dynamic ASP and then click Next.
11. Type **phonebook** in the ODBC data source name.
12. Delete all the text in the server URL input line.
13. Click Next.
14. Click Browse.
15. Browse through the directory selector to C:\InetPub\wwwroot\ phonebook.
16. Click Select.
17. Click Next to publish locally.
18. Check Yes, I Want to Create a Home Page and then click Next.
19. Click Finish. The Access Internet Publication wizard will create the site files to attach to your ODBC data source.
20. Close Microsoft Access.

Internet Database Client

Now that your database is published on the Internet, you should examine it with your Web browser. When you pull up your browser and connect to your Web site, you'll see a form that looks somewhat like the Access form you used to create it. Access automatically generates these forms using ActiveX controls that simulate the controls embedded in Access.

Browsing your site also gives you a chance to debug it. If you don't see anything at all, Active Server Pages probably can't find the ActiveX controls required to build the HTML output page. You won't get an error message because your browser gets the Web page it asked for, but the page contains no information. When Active Server Pages can't find ActiveX controls, it displays nothing, so chances are your Web pages contain references to controls that Active Server Pages can't find. Try running the Publication wizard and providing a different server URL (or none at all) to produce correct HTML output pages.

Using Web browsers and Active Server Pages to automatically create database clients that can be distributed anywhere in the world is very exciting. The Internet is finally fulfilling the real promise of client/server computing, where the computational load can be balanced on the server and on the client to minimize the amount of data sent, thus optimizing the use of high-cost long-distance network links and performing the computational workload where it makes the most sense. Because just about any computer can run a Web browser without problems, you may find that converting your database applications to run through Web browsers over an intranet is a great way to extend the life of older computers. You'll only have to upgrade one computer (the server) as the workload increases.

Unfortunately (there's always a down side), ActiveX works only on PCs running Windows and Internet Explorer. A Macintosh version is in the works, but support for other platforms such as UNIX, and for other browsers such as Netscape Navigator, will depend on the support of developers other than Microsoft—and may never develop. If you need cross-platform database client support, look to Java and the Java Database Connectivity (JDBC) standard for development. These clients unfortunately will take quite a bit more effort to develop.

Exercise 13.5 shows you how to attach to your Web site. Internet Explorer will warn you about downloading ActiveX controls unless you have configured your browser to low security level.

EXERCISE 13.5

Browsing the Phone Book Site Files

1. Launch Internet Explorer or any other Web browser.
2. Type `http://server_name/phonebook/default.html` in the address line.

3. Click the Address object.
4. Click Yes to continue if your Web browser warns you that an ActiveX control is being downloaded. You may want to disable warnings about ActiveX controls for your browsing session, as a warning will come up between each page.
5. Click the >* button at the bottom of the Web page.
6. Enter your honorific (Mr./Ms./Mrs.) in the Honorific input box and then press tab.
7. Enter your text values for the remaining field input boxes and then press tab after each.
8. Click the Commit button.
9. Double-click the C:\ drive.
10. Double-click the phonebook directory.
11. Double-click the phonebook.mdb database file.
12. Click the Forms tab.
13. Double-click the Address form.
14. Click the >| (last record) button.
15. View the record you just added through your Web browser.
16. Close Microsoft Access.

Data Access Components

Windows NT 4 Option Pack comes with a set of related services, interfaces, and components called Microsoft Data Access Components (MDAC). These components are provided to ease the development of Internet based Client/Server databases for larger databases such as SQL Server and Oracle. The core components of MDAC are:

Microsoft OLE DB is "middleware" that sits between the Web client and the server to translate standard OLE function calls into database specific calls that a certain back-end database driver (for database engines SQL Server, Oracle, or Access) will understand. Essentially, OLE DB on the server lets you use the remaining components of MDAC to build your database application.

ActiveX Data Objects (ADO) are a collection of ActiveX components that provide features to retrieve data from OLE DB interfaced databases, manipulate that data, and return data to databases. ADO objects are pre-built and can be called from VBScripts as well as compiled Visual Basic and Visual C++ applications. These objects essentially allow you to rapidly develop the ActiveX based database applications discussed in Chapter 12 without having to develop the ActiveX components yourself.

Remote Data Service (RDS) is essentially a client side data caching service that allows clients to grab complete sets of data from a server, manipulated them locally, and return them to the server. This obviates network traffic that would be involved with interactive remote manipulation, thereby speeding the process and reducing network bandwidth requirements.

The online documentation covers these products in more

detail for those who would like to build Internet database applications based on these technologies. You won't have to understand anything other than what the terms refer to for the MCSE exam. Unfortunately, the documentation that comes with these products is so jargon filled and disjointed that unless you already understand ActiveX component based development and three-tiered database development in detail you probably won't get anything useful out of it.

Microsoft Transaction Server

Microsoft Transaction Server is a *middleware* component that sits between the Internet based client and the relational database back end. MTS is a set of programming interfaces to which your database clients must be written. The MTS application itself can be run on Windows NT or Windows 95 with DCOM support. If you fail to install DCOM support for Windows 95, installations under Windows 95 will fail. Additional information about MTS is available in the online documentation.

Tip Remember that MTS can be installed on Windows NT or on Windows 95 with DCOM support.

Transaction Server provides the ability to create *transactions*, which are atomic (i.e., base-level indivisible) operations. The entire purpose of Transaction Server is to guarantee that one of two things will happen for every transaction:

- The transaction will succeed, and all components of the transaction will be correctly added to the database.
- The transaction will fail, and none of the components of the transaction will remain in the database.

This sounds simple, but it's critically important to the stability of a database system. Let's say for example that a banking transaction like a transfer between accounts consists of two operations: a deduction from one account and an addition to another account. Now, because computers can actually do only one thing at a time, one of these operations occurs before the other. If the completion of the transaction is interrupted for any reason between the two portions of the transaction, the bank computer will be left with a deduction from one account with no corresponding addition to the other. Their database will be corrupt, and you, the client, will be out that much money until the bank figures out what went wrong.

Transaction Server prevents exactly this sort of problem. By packaging a set of functions into a single atomic transaction, you are telling MTS that either all of these functions must occur or none of them can occur. Which doesn't really matter because if they fail they can simply be retried later. Due to the normal delays encountered with Web based database clients, the opportunity for failure during transactions increases dramatically, to the point that database systems would simply not be possible

without the services of a transaction system like MTS.

A Word About Database Security

Databases frequently implement security on an object level (tables, queries, stored procedures, forms, and reports may have individual security permissions) to restrict the availability of data that the server will provide. Each database system implements security differently. Some (like MS SQL Server) are well integrated with the operating system and will automatically log you on to the database using your domain credentials. Others, (like MS Access) are not built for a specific operating system and so must either log you on as an anonymous database user or ask for credentials when you open a secured database.

For Internet databases, the obvious problem is that users must be logged in correctly. For databases that are tightly integrated with the operating system, you must ensure that user coming in from the Web has properly authenticated with your server or they won't have the security permissions they expect. They'll get access denied error messages because they're logged in as an Internet anonymous user. Perhaps the easiest way to ensure proper authentication is to set security on the directory containing the Web pages that the user is initially greeted with. This will force their browser either to authenticate behind the scenes (MS Challenge and Response) or ask for credentials (Basic Authentication). Once the user is logged in properly, your database will behave as expected.

The sorts of errors you'll encounter when Internet users are not logged in correctly vary widely depending upon the database product and the specific application in use. They often show up as either database log in failures or object access errors.

Summary

Databases power modern commerce. Any serious attempt to use the Internet for direct commerce requires live access to these databases and the information they store. IIS supports attaching to databases through the Internet Database Connector and, more importantly, through Active Server Pages.

Microsoft has provided a number of database tools and utilities to allow access to databases over the Internet using Internet Information Server. The three distinct types of Internet database publications are

Static	Databases are manually published as simple HTML files.
Dynamic	HTML files containing snapshots of current database tables are automatically produced.
Interactive	Editing and updating live database information is supported over the Internet.

These three models vary in their level of usefulness and server load based upon the type of information being served.

IIS supports the ODBC interface, allowing nearly all database products to function interactively over the Internet. Microsoft Access is capable of automatically producing an entire Web site of Active Server Pages that simulate the forms used to interact with the database.

Windows NT 4 Option Pack comes with Data Access Components and Microsoft Transaction Server. Data Access Components is a suite of ActiveX objects and interfaces that allow the rapid development of Internet enabled Web clients for databases. Transaction Server provides transaction support to ensure error and corruption free database exchanges over the Internet.

Review Questions

You are running a small public service Web site detailing treatment options for athlete's foot. You have a database of health care providers and doctors in your area who specialize in the treatment of athlete's foot that you'd like to publish on your site. You update your database perhaps once every three months or so. What database publication method is most appropriate for this type of data?

- a. Static
- b. Dynamic
- c. Interactive

You've set up an offshore bookmaking operation. You'll be taking bets online using credit cards, publishing horse racing and sporting results as they become available from your online news feeds, and creating queries of winners and losers so you can credit and debit their card accounts as appropriate. The entire operation must be automatic because you expect a police raid at any time and you don't want to be present when it happens. What database publication method is most appropriate for this type of data?

- a. Static
- b. Dynamic
- c. Interactive

You've created an Interactive database using Microsoft Access and Active Server Pages, but when you browse to the site, you get garbage on your screen that looks like a computer programming language rather than the database forms you expected. What's wrong?

- a. You haven't enabled Active Server Pages output in the IIS Manager.
- b. You selected the wrong type of output in the publish to the Internet wizard in Microsoft Access.
- c. Your Web browser doesn't support Active Server Pages.
- d. You didn't set Execute permissions on the directory containing your database Web

files.

You've decided to set up an intranet database for manufacturing process reporting at your semiconductor fabrication facility. This database will extract information from many internal manufacturing control data sources and summarize it automatically so that corporate managers and top-level distributors from anywhere in the world can check current yield rates, expected inventory levels, and current manufacturing costs. What database publication method is most appropriate for this type of data?

- a. Static
- b. Dynamic
- c. Interactive

You want to set up an Internet interactive gaming site that relies on a database to store the current game state for all your connected users. You expect about 100 simultaneous users. You decide to use Microsoft Access as your database engine and Active Server Pages to interface game data in and out of the Access database. This approach means that you will only have to pay for a single Access license. This solution:

- a. Is legal and will work fine.
- b. Is legal but won't work because Access isn't powerful enough to support that many concurrent connections.
- c. Is illegal, as Access requires a client access license for all concurrent users regardless of the client software used.

You will be setting up a SQL server with an order-entry database. The database will have 200 internal data entry clerks who will be entering orders from phone calls to the company's 800 number, and 100 work-at-home data entry clerks who will be entering data from reply mail post cards. The internal data entry clerks will use a custom Visual C++ client that operates quickly (because the customer is on the phone with them) and directly attaches to the SQL server. The work-at-home data entry clerks will be connected via the Internet using IIS and an ODBC data source. How many SQL client access licenses are required?

- a. 100
- b. 200
- c. 201
- d. 300

You've set up an MS SQL Server application for work flow control and created a client based on ActiveX and Active Server Pages that runs over your company's intranet. While you developed the application on your own computer, you had no problems debugging the application. Users in the facility are also having no problems using the client through their Web browsers, but work-at-home users coming through the Internet receive an Object Access error message when they try to view database information. What is most likely wrong?

- a. SQL Server permissions are set incorrectly.

- b. IIS permissions are set incorrectly.
- c. NTFS permissions are set incorrectly.
- d. Remote users are logged in anonymously.

You've set up an MS SQL Server application for workflow control and created a client based on ActiveX and Active Server Pages that runs over your company's intranet. After a few weeks of operation, users complain that some duplicate entries are appearing in certain databases. These duplicate entries are causing some customers to be over billed. After debugging, you find that some work-at-home users are frustrated with the speed of their connection and click the Back button on their browsers to re-post information to the server when it's slow. This causes some records to be added to the database while other related records are interrupted. Which of the following solutions actually solve this problem?

- a. Re-implement the application to use Transaction Server.
- b. Increase the speed of your home users Internet links.
- c. Upgrade your server hardware so the application is more responsive.
- d. Use Data Access Components to ensure data integrity.

You are trying to determine which technology path you should take with the development of a company-wide workflow database client. All clients run Microsoft based operating systems. Shortest development time is a major consideration, and your programmers are already familiar with Visual Basic. Chose the solution with the shortest development cycle:

- a. Implement the client using ActiveX components you write and VBScript.
- b. Implement the client using ActiveX components from the Data Access Components collection and VBScript.
- c. Implement the client using Java and JDBC.
- d. Implement the client using C++.

You have an Active Server Pages program that interacts with an ActiveX control to accept query parameters and feeds them to the Internet Database Connector for a bibliography search of technical papers authored by professors at your university. The search process works great but you would like to liven up the format of the resulting output Web pages by adding some graphics and links back to your home page. Which file should you modify?

- a. Query.asp
- b. IDC.DLL
- c. Bibform.htx
- d. Query.idc

About the Authors

Matthew Strebe is an MCSE who began his career in the US Navy, installing the Navy's first fiber-optic LAN aboard a ship. He is co-author of four Network Press MCSE Study

Guides and owner of Netropolis, a network integration firm specializing in high-speed networking and Windows NT.

Charles Perkins is an MCSE with years of experience managing local and wide area networks. Co-author of four Network Press MCSE Study Guides and former Director of Computing Services for the University of Utah College of Law, he is now a consultant specializing in Windows NT.

Copyright © 1998 Sybex, Inc.

We at Microsoft Corporation hope that the information in this work is valuable to you. Your use of the information contained in this work, however, is at your sole risk. All information in this work is provided "as -is", without any warranty, whether express or implied, of its accuracy, completeness, fitness for a particular purpose, title or non-infringement, and none of the third-party products or information mentioned in the work are authored, recommended, supported or guaranteed by Microsoft Corporation. Microsoft Corporation shall not be liable for any damages you may sustain by using this information, whether direct, indirect, special, incidental or consequential, even if it has been advised of the possibility of such damages. All prices for products mentioned in this document are subject to change without notice.
International rights = English only.

International rights = English only.

Last updated May 5, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#)
Navigate
Index
Top IT Tasks [Select from this list](#)

Search TechNet
Navigate by Product

[Application Center](#)
[BizTalk Server](#)
[Commerce Server](#)
[Exchange Server](#)
[Host Integration Server](#)
[Internet Security & Acceleration Server](#)
[Office](#)
[Site Server Commerce](#)
[Small Business Server](#)
[SQL Server](#)
[Systems Management Server](#)
[Visio](#)
[Windows 2000 Professional](#)
[Windows 2000 Server](#)
[Windows 98/95/CE](#)
[Windows NT](#)
Windows Web Srvcs (IIS)

Navigate by Task
IT Solutions
Career and Training
Columns
Downloads
Troubleshoot
TechNet Community
Using TechNet
Developer

[Questions or Comments?](#)

Chapter 1 - Installing Internet Information Server

[Send to a](#)
[Print version](#)

This chapter is designed to help you install Microsoft® Internet Information Server for Windows NT™ quickly and easily.

All you need to do is connect your Windows NT Server-based computer to the Internet or your intranet (your local or wide area network), install Microsoft Internet Information Server software, and point the information server to your home directory. This chapter tells you how.

Important To publish on the World Wide Web (WWW) and the Internet, you must contact an Internet Service Provider (ISP) to obtain an Internet connection. Your ISP will provide your server's Internet Protocol (IP) address, subnet mask, and the default gateway's IP address. (The default gateway is the computer through which your computer will route all Internet traffic.)

Installation Overview

Installing Microsoft Internet Information Server is as simple as starting the Setup program on the compact disc. If you already have the necessary Internet or intranet connection, you can accept all of the default settings during setup and then add your HyperText Markup Language (HTML) content files to the \Wwwroot directory. Your files will be immediately available to users. The default setup configurations are suitable for many publishing scenarios without any further modifications.

This section defines the installation requirements and explains how to:

- Configure Windows NT before installation.
- Run the Setup program.
- Set up files to publish.
- Test your installation.

Installation Requirements

Microsoft Internet Information Server requires:

- A computer with at least the minimum configuration to support Windows NT Server; see the Windows NT Configuration and Security Checklist later in this chapter.
- Windows NT Server version 3.51 or later. Windows NT Server version 3.51 must include Service Pack 3, which is provided on the Internet Information Server compact disc.

Note Remote administration of Internet Information Server can be performed from a computer running Windows NT Workstation version 3.51 and Service Pack 3.

- Transmission Control Protocol/Internet Protocol (TCP/IP) (included with Windows NT). Use the Network applet in Control Panel to install and configure the TCP/IP protocol and related components.
- A CD-ROM drive for the installation compact disc.
- Adequate disk space for your information content. It is recommended that all drives used with Microsoft Internet Information Server be formatted with the Windows NT File System (NTFS).

[cchev] To publish on the Internet, you will need:

- An Internet connection and Internet Protocol (IP) address from your Internet Service Provider (ISP).
- Domain Name System (DNS) registration for that IP address. This step is optional, but it does allow users to use "friendly names" instead of IP addresses when connecting to your server. For example, microsoft.com is the domain name registered to Microsoft. Within the microsoft.com domain, Microsoft has named its World Wide Web (WWW) server www.microsoft.com. Most ISPs can register your domain names for you.
- A network adapter card suitable for your connection to the Internet.

[cchev] To publish on an intranet, you will need:

- A network adapter card and Local Area Network (LAN) connection.
- The Windows Internet Name Service (WINS) Server or the Domain Name System (DNS) service installed on a computer in your intranet. This step is optional, but it does allow users to use friendly names instead of IP addresses.

Windows NT Configuration and Security Checklists

Before installing Microsoft Internet Information Server, you must configure the Windows NT Server networking component so that your server can operate on the Internet. You may want to also enhance the Windows NT Server default security settings and implement other Windows NT security measures to prevent Internet users from tampering with your computer or network. For more information about security, see Chapter 5, "Securing Your Site Against Intruders."

Windows NT Configuration Checklist

Use the Network applet in Control Panel for all configuration tasks mentioned in this section.

- **Obtain an Internet Connection.** To publish on the Internet, you must have a connection to the Internet from an Internet Service Provider (ISP). To find an ISP, look in the telephone book under Computers-Networking or in your local newspaper's business or technology section.
- **Install Windows NT Server.** Install Windows NT Server version 3.51 and Service Pack 3 (Service Pack 3 is included on the Internet Information Server compact disc).
- **Configure the TCP/IP Protocol.** Install the Windows NT TCP/IP Protocol and Connectivity Utilities. If the FTP Service provided with Windows NT has been installed, remove it. Your ISP must

provide your server's IP address, subnet mask, and the default gateway's IP address. (The default gateway is the ISP computer through which your computer will route all Internet traffic.)

- **Configure the Server's Domain Name** (also called Host Name). Your IP address (for example, `http://10.138.59.1/homepage.htm`) can always be used to contact your Web server. However, if you register a domain name in the Domain Name System (DNS), your server can be contacted by using a "friendly" domain name (for example, `http://www.company.com/homepage.htm`). ISPs can usually register domain names for you.
- **Configure Name Resolution.** You must have a Domain Name System (DNS) server's IP address in order to use its friendly name in Internet Explorer when browsing other servers (and your own server, if it is registered in DNS) on the Internet.

If your server will be on a TCP/IP LAN with WINS servers, obtain the name of the WINS server to use.

An alternative to DNS is to use a HOSTS file. On intranets an alternative to WINS Servers is to use an LMHOSTS file. Make the appropriate Advanced TCP/IP Configuration setting for this server's name resolution.

- **WWW Virtual Servers** Optionally, if you have registered multiple domain names (such as `www.company1.com` and `www.company2.com`), you can host multiple domain names on the same computer running Microsoft Internet Information Server. You use Advanced TCP/IP Configuration settings to assign multiple IP addresses to the network adapter card connected to the Internet. You should register a domain name for each IP address on your adapter.

If you need to add more than five virtual servers, see Help for more information.

Windows NT Security Checklist

Several steps can be taken to enhance the security of a computer connected to the Internet. For further information on these checklist topics, see Chapter 5, "Securing Your Site Against Intruders."

User Accounts

- Review the `IUSR_computername` account's rights.
- Choose difficult passwords.
- Manage strict account policies.
- Limit the membership of the Administrators group.

NTFS File Security

- Use NTFS.
- Enable Auditing.

Running Other Network Services

- Run only the services that you need.
- Unbind unnecessary services from your Internet adapter cards.

- Check permissions on network shares.

Contents of the Compact Disc

The compact disc contains the following directories:

/<root>

Use Setup.exe in the root directory to install Microsoft Internet Information Server and all components.

/Admin

Use Setup.exe in this directory to install Internet Service Manager only.

/Alpha

Contains the files to install Internet Information Server on an Alpha AXP™ processor.

/Clients

Use Setup.exe to install Internet Explorer only.

/Help

Contains WinHelp files.

/I386

Contains the files to install Internet Information Server on an Intel® processor.

/Mips

Contains the files to install Internet Information Server on a MIPS® processor.

/Ppc

Contains the files to install Internet Information Server on a PowerPC® processor.

/Samples

Contains sample HTML content files.

/Sdk

Contains some of the header files for the Internet Extensions for Win32 and Internet Server Application Programming Interface (ISAPI). Also contains .MIB files to use with SNMP monitoring. (A complete ISAPI Software Development Kit is also available from Microsoft.)

Winnt351.qfe

Run Update.exe in this directory to install Service Pack 3.

How to Install Internet Information Server

Once you have your Internet connection and have configured Windows NT Server, you can install Microsoft Internet Information Server. This section tells how to start the Microsoft Internet Server Setup program from the compact disc.

To install the Internet Information Server services, you must be logged on with administrator privileges. In addition, to configure the Internet Information Server services by using the Internet Service Manager, your user account must be a member of the Administrators group on the target computer.

[cchev] To start Setup

1. Insert the Microsoft Internet Information Server compact disc into an appropriate drive.
2. In File Manager or at the command prompt, change to the drive containing the compact disc.

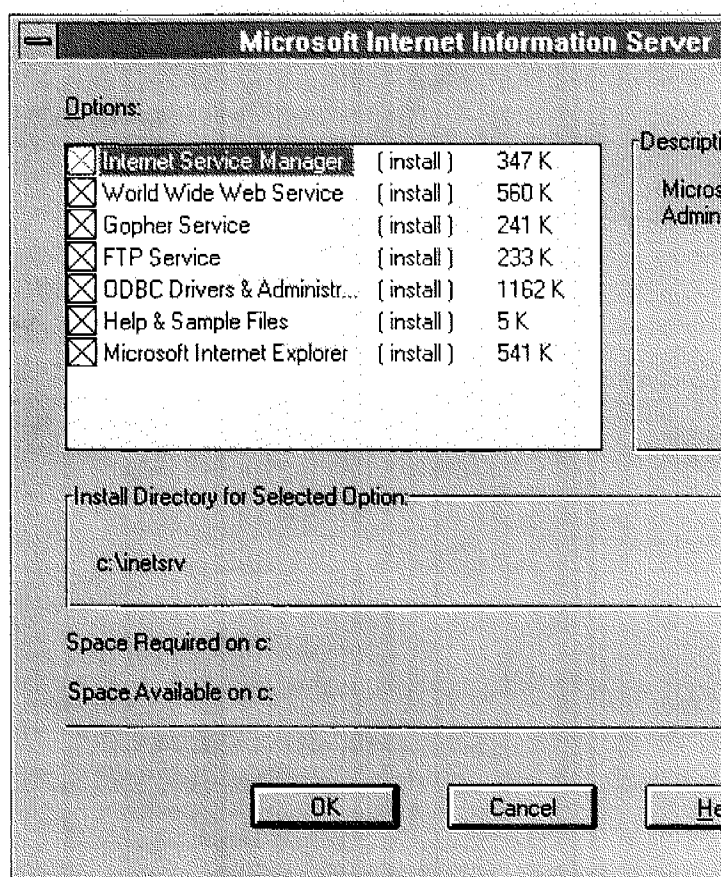
Start Setup:

- To start Setup from File Manager, double-click the file named Setup.exe in the root directory of the compact disc.
- To start Setup from the command prompt, change to the root directory of the compact disc and then type **setup**.

Note During setup, you can choose the Help button in any dialog box to get assistance. When you do, a Help topic is displayed that explains the choices you have at that point and the procedure to complete the dialog box.

3. If you have not installed Service Pack 3 for Windows NT version 3.51, a dialog box will appear and offer to install the service pack automatically at the conclusion of Setup. Microsoft Internet Information Server will not operate without Service Pack 3 installed. Choose the Yes button to install Service Pack 3. Note that at the conclusion of the Service Pack Update you must restart your computer.
4. The Microsoft Internet Information Server Welcome dialog box appears. Choose the OK button.

The second dialog box appears, displaying the following installation options:



5. All of the following items are selected for installation by default. If you do not want to install a particular item, click the box next to it to clear it.

Internet Service Manager installs the administration program for managing the services.

World Wide Web Service creates a WWW publishing server.

Gopher Service creates a Gopher publishing server.

FTP Service creates an FTP publishing server.

ODBC Drivers and Administration installs Open Data Base Connectivity (ODBC) drivers. These are required for logging to ODBC files and for enabling ODBC access from the WWW service.

Important If you want to provide access to databases through the Microsoft Internet Information Server, you will need to set up the ODBC drivers and data sources by using the ODBC applet in Control Panel. Please see Chapter 8, "Publishing Information and Applications" for specific instructions.

If you have an application running that uses ODBC, you may see an error message telling you that one or more components are in use. Before continuing, close all applications and services that use ODBC.

Help and Sample files installs online Help and sample HyperText Markup Language (HTML) files.

Microsoft Internet Explorer installs the Web browser, Microsoft Internet Explorer.

You can use the Setup program later to add or remove components. Setup can also be used to remove all Internet Information Server components.

6. Accept the default installation directory (C:\Inetsrv) or click the Change Directory button and enter a new directory.

Note If you have installed Internet Information Server, but want to reinstall it into another directory, you must remove the following key from the Registry:
\\HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\INetSrv. If you do not delete this key, the Change Directory button will be dimmed and you will be unable to change the default directory.

7. Choose the OK button.

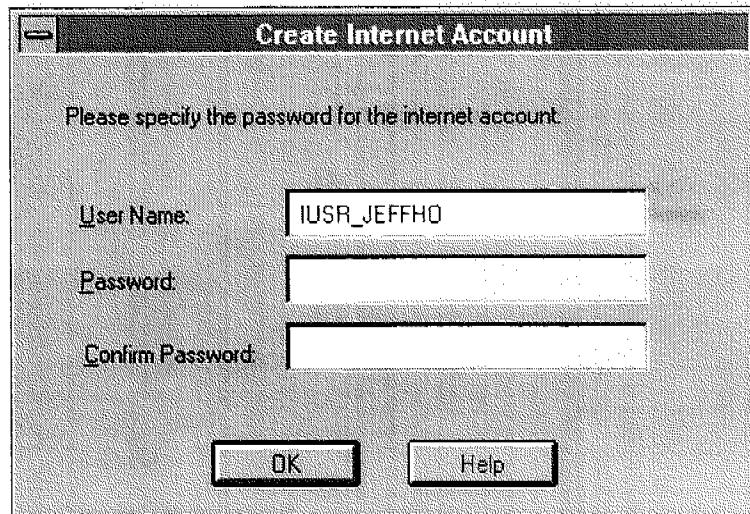
The Publishing Directories dialog box appears.

Accept the default directories for the publishing services you have installed, or change the directories.

Note If you already have files ready to publish, you can enter the full path to their current location, or move them into the default directories later. If your files are on a network drive, you should accept the default directory. After setup is completed, use Internet Service Manager to change your default home directory to the path for the network directory containing your files; for example, \\Servername\\Sharename\\WWWfiles. Be sure to carefully check the permissions on the network drive; there may be security implications. See Chapter 5, "Securing Your Site Against Intruders."

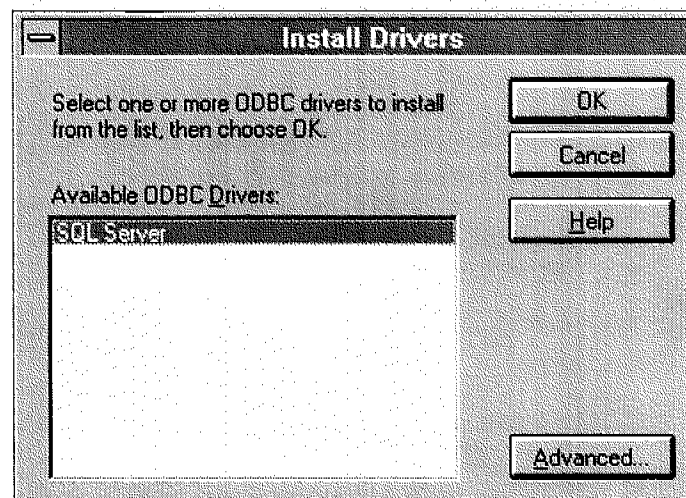
8. Choose the OK button.

9. When prompted to create the service directories (Wwwroot, Gophroot, and Ftproot by default), click Yes.
10. The Create Internet Account dialog box appears. This is the account used for all anonymous access to the Internet Information Server. You should enter a password and confirm the password for this account. Choose OK.



The 'Create Internet Account' dialog box has a title bar with a minus button and the text 'Create Internet Account'. Below the title bar, it says 'Please specify the password for the internet account.' There are three input fields: 'User Name:' with the text 'IUSR_JEFFHO', 'Password:', and 'Confirm Password:'. At the bottom, there are two buttons: 'OK' and 'Help'.

11. Setup copies all remaining Internet Information Server files.
12. If the ODBC Drivers and Administration option box was selected, the Install Drivers dialog box appears.



The 'Install Drivers' dialog box has a title bar with a minus button and the text 'Install Drivers'. Below the title bar, it says 'Select one or more ODBC drivers to install from the list, then choose OK.' There is a list box labeled 'Available ODBC Drivers:' containing 'SQL Server'. To the right of the list box are three buttons: 'OK', 'Cancel', and 'Help'. At the bottom right, there is an 'Advanced...' button.

To install the SQL Server driver, select the SQL Server driver from the Available ODBC Drivers list box, and choose the OK button.

Setup completes copying files.

13. The Setup completion dialog box appears. Click the OK button to complete Setup.
14. If, during Setup, you were prompted to install the Service Pack 3 update and you answered Yes, the Service Pack 3 update program will start automatically after setup. At the conclusion of the update, you must restart your computer.

The preceding steps are all that is required for a simple installation. You are now ready to publish on the Internet or your intranet. There is no need to start Internet Service Manager unless you want to make advanced configuration changes. (If so, refer to Chapter 3, "Configuring and Managing Your Internet Information Server.") Use the Services applet in Control Panel to confirm successful installation of the World Wide Web publishing service.

How to Install Internet Explorer or Internet Service Manager

The compact disc contains two additional Setup.exe programs that will allow you to install Internet Explorer only or Internet Service Manager only. These Setup programs are nearly identical to the Setup program described above; however, only the relevant options are available in the setup options dialog box.

To install Internet Explorer only, use Setup.exe in the \Clients directory on the compact disc. To install Internet Service Manager only, use Setup.exe in the \Admin directory on the compact disc. Follow the directions on the screen. For more information see Help or refer to the previous section in this chapter.

These Setup programs are most useful if you will use Internet Information Server on a local area network (LAN). You can copy the contents of the \Clients directory to a shared network directory to enable clients to install Internet Explorer over the network. You can copy the contents of the \Admin directory to a shared network directory to enable administrators to install Internet Service Manager for remote administration of Internet Information Server from any computer on the network running Windows NT Workstation or Windows NT Server.

Unattended Setup When Installing from a Network Directory

If you are using Microsoft Internet Information Server on a network you can copy the contents of the compact disc to a directory on your network and perform unattended installations over the network from that directory. (You can start an unattended setup from the compact disc itself; however, only the default configuration can be installed in this case.) This is useful for installing several servers at your site or to provide a simple over-the-network installation process for Internet Explorer users.

In each directory containing a Setup.exe file is the file Unattend.txt. Unattend.txt is a sample configuration file used by the program for unattended installation. You modify the values in the file to configure setup. In general, the value 1 represents TRUE and the value 0 represents FALSE. It is suggested that you copy Unattend.txt to the directory containing the Setup.exe you will use, then modify it to meet your installation requirements.

To start unattended mode setup you must use the command prompt. Change to the directory containing both Setup.exe and Unattend.txt and type

setup -b unattend.txt

where Unattend.txt is the file you have modified. See Unattend.txt on the compact disc for more information about unattended setup.

The IUSR_computername Account

Setup automatically creates an account called IUSR_*computername* (where *computername* is the computer name specified in the Network applet of Control Panel). This account has an empty password and privilege to log on locally. On domain controllers, this account is added to the domain database.

Use the Service property sheet in Internet Service Manager to change the user account used on behalf of all remote clients that log on with the anonymous IUSR_*computername* account.

Use User Manager to add a password to the account, then specify the password for IUSR_*computername* by using Internet Service Manager. You should carefully review the rights granted to the account used, including the default IUSR_*computername* account created during setup.

How to Publish Information

Now that Microsoft Internet Information Server is installed and running, you are ready to publish on the Internet or your intranet. Providing information with Internet Information Server is easy. If your files are in HTML format, just add them to the appropriate home directory. For example, if you are using the WWW service, place the files in the \Wwwroot directory.

For more extensive information on creating and publishing content files, see Chapter 8, "Publishing Information and Applications." Note that you can also create and publish highly interactive systems by writing programs using ISAPI.

Note If you provide files with the Gopher or File Transfer Protocol (FTP) services, you can share those files instantly. Users can navigate through the files much as they do in File Manager or at the command prompt. With Gopher, you can customize how your directories and files appear to browsers; you can also include links to other servers in your files. FTP can be used to accept files from or send files to Internet users.

How to Test Your Internet Information Server Installation

You can test your installation by using Internet Explorer to view the files in your home directory.

[cchev] To test a server connected to the Internet

1. Ensure that your server has HTML files in the \Wwwroot directory.
2. Start Internet Explorer on a computer that has an active connection to the Internet. This computer can be the server you are testing, although using a different computer is recommended.
3. Type in the Uniform Resource Locator (URL) for the home directory of your new server.

The URL will be "http://" followed by the name of your server, followed by the path of the file you want to view. (Note the forward slash marks.) For example, if your server is registered in DNS as "www.company.com" and you want to view the file "homepage.htm" in the root of the home directory, in the Location box you would type:

http://www.company.com/homepage.htm

then press the ENTER key. The home page should appear on the

screen.

[cchev] To test a server on your intranet

1. Ensure that your computer has an active network connection and that the WINS server service (or other name resolution method) is functioning.
2. Start Internet Explorer.
3. Type in the Uniform Resource Location (URL) for the home directory of your new server.

The URL will be "http://" followed by the Windows Networking name of your server, followed by the path of the file you want to view. (Note the forward slash marks.) For example, if your server is registered with the WINS server as "Admin1" and you want to view the file "homepage.htm" in the root of the home directory, in the Location box you would type:

http://admin1/homepage.htm

Then press the ENTER key. The home page should appear on the screen.

Last updated January 12, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#) |**Navigate**[Index](#)[Top IT Tasks](#) | [Select from this list](#)[Search TechNet](#)**Navigate by Product**

[Application Center](#)
[BizTalk Server](#)
[Commerce Server](#)
[Exchange Server](#)
[Host Integration Server](#)
[Internet Security & Acceleration Server](#)
[Office](#)
[Site Server Commerce](#)
[Small Business Server](#)
[SQL Server](#)
[Systems Management Server](#)
[Visio](#)
[Windows 2000 Professional](#)
[Windows 2000 Server](#)
[Windows 98/95/CE](#)
[Windows NT](#)
[Windows Web Srvcs \(IIS\)](#)

Navigate by Task**IT Solutions****Career and Training****Columns****Downloads****Troubleshoot****TechNet Community****Using TechNet****Developer**[Questions or Comments?](#)

Chapter 4 - Networking for the Internet or an Intranet

The Internet is a network of networks. Every Internet Information Server must be configured to operate in a network, whether it is the global Internet or your local intranet.

This chapter explains:

- Routers and security devices.
- Typical network configurations.
- Administering servers by using Internet Service Manager.
- Using the discovery mechanism to find other computers on your network.
- Internet publishing requirements
- Issues involved in publishing on a private intranet.
- Internet Explorer for network users.
- Using Simple Network Management Protocol (SNMP) monitoring.

General Networking Issues

This section explains the basic Transport Control Protocol/Internet Protocol (TCP/IP) networking requirements for nearly all Internet Information Server sites, especially those with more than one information server. For issues specific to the Internet or intranet publishing, see those sections later in this chapter.

Routers and Security Devices

TCP/IP is a routeable protocol, meaning each piece of information (packet) has a specific address that it is routed to. Dedicated routers connect two networks, routing packets between the networks. The routers check the destination for each packet on one network, and if the destination is on the router's other network, it routes the packet to its destination.

Routers can be configured to allow only certain packets between networks, a process called packet filtering. Packet filtering can be used to prevent users from seeing or connecting to internal computers and resources.

If you have a TCP/IP network you probably have routers in your network already. Often an Internet Service Provider (ISP) will install a router between the Internet and your information server. This will enable you to filter the incoming and outgoing packets. See your ISP or router documentation for more information about configuring routers or similar security devices.

Typical Network Configurations

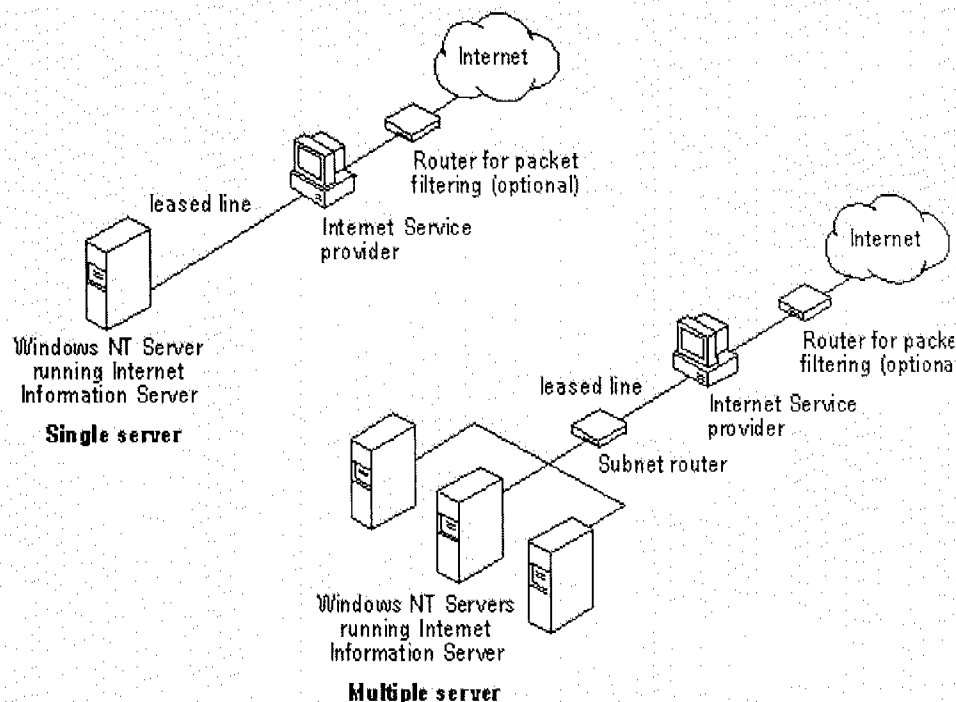
Network configuration is based on whether you will have an Internet site or an intranet site.

Internet Sites

If you will have only one computer running Internet Information Server at your site, your Internet Service Provider (ISP) can help you with many details, such as route configuration and the IP address of the default gateway that your server will use.

If you have multiple computers running Internet Information Server on your network, you must configure their TCP/IP settings to operate correctly through your Internet connection configuration, including any routers used between your servers and the default gateway.

Typically, sites with more than one computer running Internet Information Server will add another router. With the addition of another router, the servers can be grouped into a single subnet isolated from your private network, as shown in the following diagram.



[cchev] To create a subnet you will need:

- One computer with two network adapter cards and Windows NT TCP/IP routing enabled, or a dedicated router for your subnet.

See Help in Windows NT for the procedure to create a simple router on a computer running Windows NT and for the procedure to set routing tables by using the **route** command.

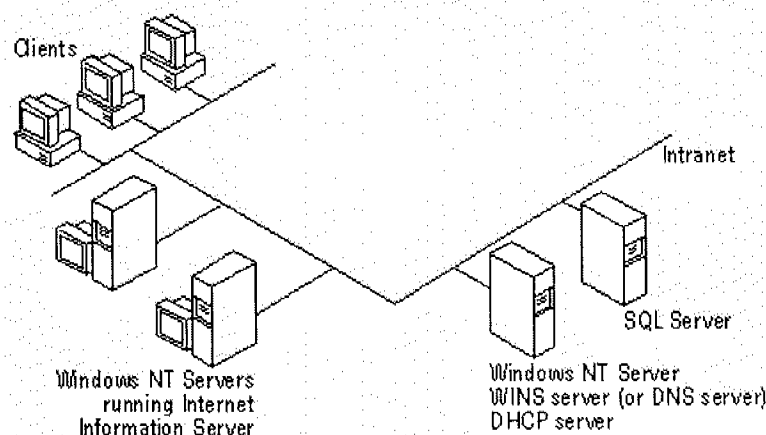
- Valid IP addresses for every network adapter card in your subnet and the correct Subnet Mask.
- Correct Default Gateway IP address configurations.

Your ISP will provide you with the Internet IP addresses, subnet mask (if any), and your default gateway configuration.

Intranet Sites

If you are publishing only to your own intranet, Internet Information Server can be integrated into any TCP/IP network. If Dynamic Host Configuration Protocol (DHCP

and Windows Internet Name Service (WINS) are enabled on your network, clients can use the server's computer name to connect with the server. If Domain Name System (DNS) is enabled on your network, you will use host names.



Integrating Your Intranet with the Internet

It is possible to just connect your entire intranet to the Internet, rather than connecting a subnet containing only your Information Servers to the Internet. However, there are many security implications to connecting an intranet to the Internet. You should thoroughly understand the security implications and understand TCP/IP networking before you decide to integrate your entire network with the Internet. Integrating a network with the Internet requires information that is outside the scope of this manual. See Chapter 5, "Securing Your Site Against Intruders," for more information about security, and consult the Internet or other sources for additional information about Internet security, firewalls, and TCP/IP networking.

Administering Servers with Internet Service Manager

You can install Internet Service Manager on computers from which you will administer computers running Internet Information Server on your network. Internet Service Manager can be installed on Windows NT Workstation or Windows NT Server.

Note All Internet Information Server services (WWW, Gopher, and FTP) require Windows NT Server. Internet Service Manager can also be installed on computers running Windows NT Workstation.

For over-the-network installation, use File Manager to create a network share containing the \Admin directory on the compact disc. You can then install Internet Service Manager to administer the services from any computer on the network running Windows NT version 3.51.

Finding Other Computers on Your Network or Subnet

Microsoft Internet Service Manager has a discovery mechanism that finds computers running Microsoft Internet services on your network. You can choose Find All Servers in the Properties menu to discover the Microsoft Internet Information Server computers on your network.

If WINS servers are used on your network, the discovery process used by Microsoft Internet Server is automatic. When Microsoft Internet Information Server starts, it automatically registers its available services with your WINS servers. Thus, when Internet Service Manager queries the network for computers running Microsoft Internet services, the WINS servers return the registered services. Internet Service Manager then displays the returned services.

If WINS servers are not available, discovery uses TCP/IP broadcasts to perform the same functions. Discovery will not work if you do not have WINS servers, or if the servers reside across routers and cannot be discovered by using broadcasts.

Publishing on the Internet

For the world to reach your site, you must have an Internet connection. Connections to the Internet are usually leased from ISPs. In addition to providing your physical Internet connection and IP address (and subnet mask if appropriate), your ISP can provide many of the Internet services, such as domain name registration, routers, and DNS service.

How to Choose the Right Internet Connection

Your connection to the Internet will be through a network adapter card or other network device, such as a modem or Integrated Services Digital Network (ISDN) card. Internet bandwidth is measured in bits per second (bps).

Your server configuration and Internet bandwidth determine how fast data gets to your computer and how many requests can be serviced simultaneously. As the number of computers getting data through your Internet connection increases, delays or failures will occur unless you have enough bandwidth.

When you lease an Internet connection a network cable is installed by your ISP to your site. Leased connection speeds range from 56,000 bps (with Frame Relay) to 45,000,000 bps (with a T3 connection). A dial-up ISDN line can offer speeds up to 128,000 bps.

Internet Connection Types

The connection types described in the following table represent typical levels of service for full Internet connections. (Some ISPs provide only limited Internet service.) The Internet services offered through Internet service providers in your area may differ slightly.

Connection Types

Connection	Maximum BPS	Simultaneous Users Supported
Frame Relay	56,000	10-20
ISDN	128,000	10-50
T1	1,500,000	100-500
Fractional T1	varies as needed	
T3	45,000,000	5000+

A light-duty server can use Frame Relay or ISDN. A server with medium traffic might have a T1 line or some fraction of a T1 line installed. Large businesses that expect heavy Internet traffic may need fractional or multiple T1 lines or even T3 service in order to handle thousands of users.

Modem connections to the Internet are available, but are typically used for individual client browsing, and are not recommended for servers. A connection to the Internet using a phone line and modem can service only two or three simultaneous users. (Modem connections might be used for text-only Internet servers with only a small number of potential users.) Modem connections are often called "slow links" because data is transmitted at the speed of the modem, typically from 9,600 to 28,800 bps, far too slow for efficient operation of a WWW server.

IP Addresses and DNS

The Internet is a world-wide collection of individual Transmission Control Protocol/Internet Protocol (TCP/IP) networks. Each computer on the Internet has a

unique address (IP address). Information is transmitted on the Internet in data packets. Each packet is addressed to a specific computer's IP address, such as 10.212.57.189.

Because IP addresses are difficult to use and remember, the Domain Name System (DNS) was created to pair a specific IP address, such as 10.189.54.1, with a friendly domain name, such as microsoft.com. When a user browses the Internet by using domain name, the browser first must contact a DNS server to resolve the domain name to an IP address, then contact the computer with that address.

This has two implications for your Internet Information Server:

- You must have a permanent IP address assigned to a server on the Internet
- You must register a domain name in the DNS for your permanent IP address

Your ISP will generally provide your IP addresses and may also register your domain names. Contact the Internet Network Information Center (InterNIC) or your ISP for more information about DNS registration.

Other Internet Client Services

Your ISP must provide you with a connection, one or more IP addresses (and subnet mask, if appropriate), and usually the IP address of at least one DNS server. Internet service providers often offer additional client services. You will need additional software to use these services.

Mail services are used to exchange electronic mail. The Simple Mail Transfer Protocol (SMTP) is used for Internet mail.

News services give you access to a Network News Transfer Protocol (NNTP) server. Using a news reader, you can read messages posted in the thousands of available news groups. Usenet is one of the more popular public news services.

Publishing on an Intranet

Microsoft Internet Information Server can also be used on any private TCP/IP network to provide files and applications to network users. This section explains how to plan for publishing on a private intranet. Issues to be considered include:

- Distributing Internet Explorer to Clients.
- Name Resolution Systems.
- Using DHCP.
- Using Computer names in URLs.

Internet Explorer

Internet Explorer makes it easy for users to browse your information services. Use point and click on links to move from page to page. If links to nonHTML files are encountered, Internet Explorer automatically displays the file with the proper view or downloads the file to the local hard drive.

Internet Explorer versions are included for your intranet users running any of the following operating systems:

- Windows NT Server version 3.51 or later
- Windows NT Workstation version 3.51 or later
- Windows 95

- Windows for Workgroups version 3.11
- Windows version 3.1

What are the Differences Between Versions?

All Internet Explorer versions perform the same basic functions and have very similar operation. Internet Explorer takes advantage of the features of the operating system it is running. Setup.exe in the \Clients directory on the compact disc automatically installs the correct version.

Windows NT Server and Windows NT Workstation

This version of Internet Explorer runs on versions 3.51 and later. It is a 32-bit application.

Windows 95

This version of Internet Explorer runs on Windows 95. It is a 32-bit application that takes advantage of the Windows 95 interface.

The Windows95 version of Internet Explorer also supports many advanced features, such as:

- Inline video (.avi files)
- Background sound and bitmaps
- Scrolling banners
- Context-sensitive menus

Windows for Workgroups and Windows version 3.1

This version of Internet Explorer runs on Windows for Workgroups version 3.11 and Windows version 3.1. It is a 16-bit application.

How Do I Distribute Internet Explorer to Users?

You can use File Manager to share the contents \Clients directory on your compact disc, and then instruct users to run the Setup program from the network share. Setup automatically installs the appropriate version.

You can also copy the \Clients directory to a network share on a hard disk and allow clients to run Setup from the network share.

To fully automate installation for clients and control the installation configuration, you can use the file Unattend.txt. Unattend.txt is in each directory containing Setup.exe. First modify Unattend.txt to reflect the default configuration for users, then instruct users to install Internet Explorer from a batch file that starts unattended-mode Setup. See Chapter 1, "Installing Internet Information Server," Help for more information about unattended-mode setup.

Name Resolution Systems

If you want intranet clients to be able to use friendly names with Internet Explorer when browsing information servers, you must provide a name resolution system for clients.

Windows NT Server offers you the advantage of automatic IP address administration with the DHCP server and WINS server methods for name resolution offered by WINS servers.

Using Computer Names with WINS Servers

A WINS server is a Windows NT Server-based computer running Microsoft TCP/IP and WINS server software. A WINS server maintains a database that maps TCP/IP addresses to Windows Networking computer names.

Microsoft Internet Information Server uses WINS server software to map TCP/IP addresses to computer names on the network. WINS uses Microsoft Networking computer names, which makes it much more flexible than DNS for name resolution. WINS also provides a dramatic reduction of IP broadcast traffic in Microsoft internetworks, while allowing client computers to easily locate remote systems across local or wide area networks. If you use WINS servers on the Internet, your computers must be using valid Internet IP addresses.

Using Computer Names and LMHOSTS

An LMHOSTS file is a simple text file resolving Windows computer names to IP addresses. If you have a small or infrequently changing network you can distribute an LMHOSTS file to each computer in the network. Each time a host changes you have to manually change the LMHOSTS files.

Using Domain Names with DNS Servers

You can maintain a DNS server and Internet-assigned TCP/IP domain names as us on the Internet. If you plan to connect your network to the Internet, your IP addresses and DNS server routing configuration must be valid for the Internet.

Using Domain Names and HOSTS

A HOSTS file is a simple text file resolving DNS domain names to IP addresses. If you have a small or infrequently changing network, you can distribute a HOSTS file to each computer. Each time a host changes you will have to manually change the HOSTS files.

Using DHCP in Your Intranet

You can take advantage of DHCP server automatic IP address administration.

A DHCP server is a Windows NT Server-based computer running Microsoft TCP/IP and the DHCP server software.

If you use DHCP servers, you must use WINS Servers for clients to have automatic IP address name resolution. DHCP is defined in Requests for Comments (RFCs) 1533, 1534, and 1541. See Tcip.hlp in Windows NT Server for more information about DHCP servers.

Using URLs and Creating HTML Links for Intranets

When you connect to a server or create HTML files and links on an intranet, you must name computers in accordance with the name resolution system implemented on your network. For example, if you use WINS servers on your network, your link will use Windows computer names, such as <http://sales1/homepage.htm>, where sales1 is the name of the computer running Internet Information Server.

SNMP Monitoring

If you monitor your network by using Simple Network Management Protocol (SNMP) you can use the SNMP Management Information Bases (MIBs) provided by Microsoft Internet Information Server to monitor your Web server.

The MIB files included in the \Sdk directory of the Microsoft Internet Information Server compact disc can be used by third-party SNMP monitors to enable SNMP monitoring of the WWW, Gopher, and FTP services of Microsoft Internet Information Server.

Internet Information Server supports SNMP monitoring only. SNMP configuration is not supported.

You will need to compile the MIB files using the MIB compiler that comes with your SNMP software before using them with the Windows NT SNMP service. You must start the services to be monitored before configuring and starting the SNMP service on your Internet Information Server-based computer. Once the SNMP service has been started on both the remote and local computers, you can use SNMP tools to monitor the running services.

See Help for more information about SNMP.

last updated January 12, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#) |[Navigate](#) | [Index](#) | [Top IT Tasks](#) | [Select from this list](#) | [go](#) | [Search TechNet](#)**Navigate by Product**

- Application Center
- BizTalk Server
- Commerce Server
- Exchange Server
- Host Integration Server
- Internet Security & Acceleration Server
- Office
- Site Server Commerce
- Small Business Server
- SQL Server
- Systems Management Server
- Visio
- Windows 2000 Professional
- Windows 2000 Server
- Windows 98/95/CE
- Windows NT
- Windows Web Svcs (IIS)**

Navigate by Task**IT Solutions****Career and Training****Columns****Downloads****Troubleshoot****TechNet Community****Using TechNet****Developer**[Questions or Comments?](#)

Web Server Configuration

Leonid Braginski and Matthew PowellChapter 4 from *Running Microsoft Internet Information Server*, published by Microsoft Press [Send this document to a colleague](#) [Printer-friendly version](#)

Topics on this Page

- ▼ Setup
- ▼ The IIS Hierarchy
- ▼ Configuration Options

The heart of Microsoft Internet Information Server (IIS) is the World Wide Web publishing service. In Chapter 12 you will learn about services in detail. For now, it is enough to know that a *service* is a special program that runs in the background and performs certain tasks. The IIS Web server not only receives HTTP requests but also activates other components such as Active Server Pages. The Web Publishing Service, for example, handles Web browser requests by sending requested Web pages back to the browsers. This chapter takes a quick look at how the Web server is installed and discusses the many configuration options available in this environment.

Setup

At the beginning of the installation process, the IIS Setup provides both minimal and typical installation options, in addition to a custom option. Minimal installation is designed to conserve your hard drive space—only bare necessities are installed. Typical installation adds some optional components such as Microsoft Script Debugger as well as extended documentation. The Custom installation option lets you choose installation options from a list of all available components.

If you are interested in running only WWW service and have limited hard disk space, the Minimal installation option might be the best one. With the Minimal installation option, Setup creates different directories during installation, including the \InetPub directory that will hold much of your published information. Two subdirectories included in \InetPub are \scripts, which is usually reserved for ISAPI, CGI, and ASP, and \wwwroot, which is the default home directory for the Web site. You'll learn much more about ASP applications in Chapter 17 and about ISAPI and CGI applications in Chapter 20.

Note Throughout this book, whenever a particular component is discussed, we will point out which installation option you need to choose to install it.

The Typical installation adds additional components, directories, and documentation to your hard drive, but the basic directory structure stays the same.

If you choose the Custom installation option, you will be presented a list of choices during IIS installation. Figure 4.1 shows the Microsoft Windows NT 4.0 Option Pack Setup window when the Custom installation option is selected.

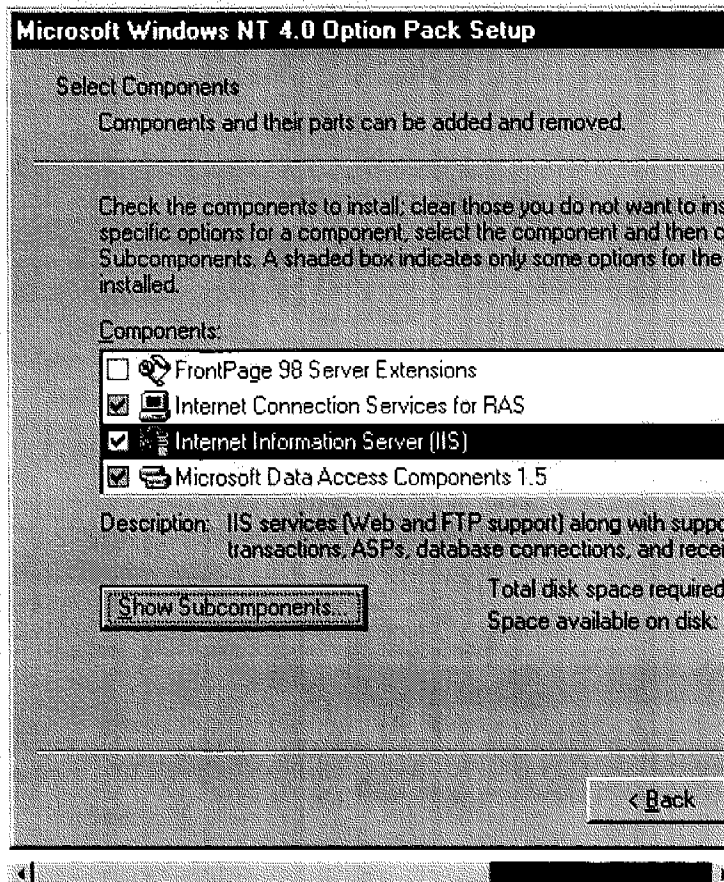


Figure 4.1 The components of the Windows NT Option Pack with the Internet Information Server (IIS) component selected.

After the user selects a component to install and clicks the Show Subcomponents button, the dialog box shown in Figure 4.2 appears, listing the specific subcomponents for the component selected (in this case, IIS). Most of the subcomponents listed in this dialog box represent additional services that run alongside IIS. These services will be discussed in later chapters.

Note The only optional subcomponent directly related to the Web Publishing Service (WWW service) is the installation of the HTML version of the Internet Service Manager. The HTML Internet Service Manager (which we will call HTML-based administration pages) allows you to administer your WWW service from any browser (subject to secure authentication).

In Figure 4.2, the HTML Internet Service Manager subcomponent is checked so that it will be installed.

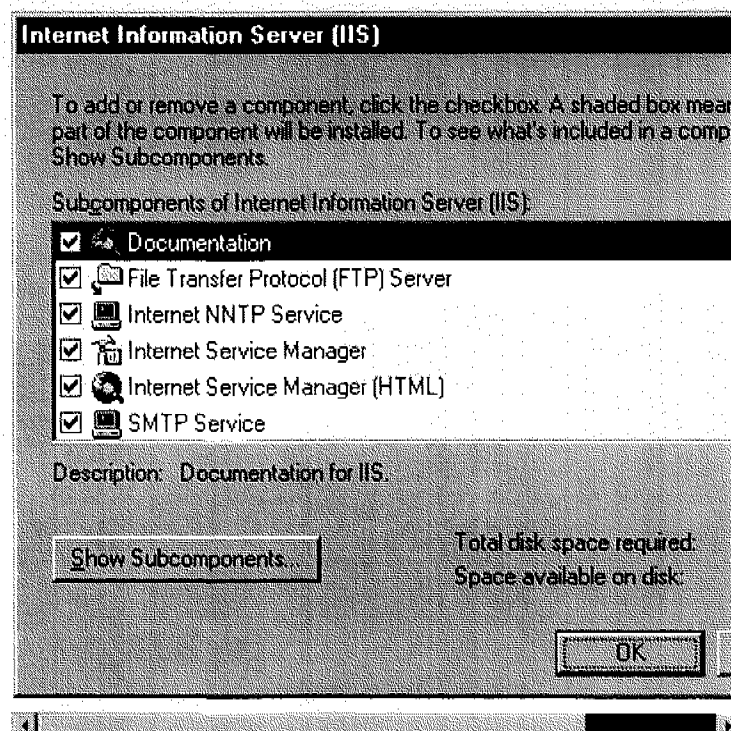


Figure 4.2 The subcomponents of Internet Information Server.

The IIS Hierarchy

Before you explore the configuration options available under IIS, you need to understand a few concepts, including the hierarchy of the following IIS configurable units: the Web server, Web sites, home directories, virtual directories, and applications. The Web server, the top level of our hierarchy, can contain multiple Web sites on a single machine. A Web site is usually considered the equivalent of a single host machine running a Web server. However, using IIS, multiple Web sites can be located on a single machine.

Web Sites

If you choose to install the optional HTML Internet Service Manager (which we will call HTML-based administration pages), two Web sites will automatically be created on your machine: the default site and the administration Web site. Most standard HTTP requests are handled at the default site. The administration site is waiting to accept connections from browsers on a nonstandard TCP (Transmission Control Protocol) port (other than 80). The sole purpose of the administration site is to provide remote administration capabilities over HTTP protocol. Therefore, it is configured as an entity in itself.

As noted earlier, IIS allows you to have multiple sites on one machine. That means you can create other sites that listen on different ports or use other IP addresses or even sites that just use a different host name. For example, a single machine could have a site created for the host name *www.cmpny.com* and another site created for *www.cmpny2.com*.

Web sites can contain many virtual directories and applications. Each individual site can have its own distinct properties, such as directory structure, default documents, scripts, and so on. For example, even though both *http://www.cmpny.com* and *http://www.cmpny2.com* are located on the same physical machine, the two sites could be completely different. By locating different sites on the same machine, you can access many virtual computers for the cost and maintainability of one physical machine.

The Home Directory

Each site on a machine must have one and only one home directory. The home directory is the site's default location on the server and contains the default page shown in the browser when the server is accessed without being asked for a specific Web page. Assume that the browser specifies a URL such as *http://www.microsoft.com*. Because no specific Web page is requested in this URL, the Web server finds a default document in the home directory (DEFAULT.HTM, for example) and sends it to the browser. You cannot change or delete a site's home directory, although it is possible to configure a site's home directory as a different physical directory on another computer on the same network. (You will learn how to do this later in this chapter.)

The Virtual Directory

The IIS Web server does not make just any directory available to client browsers—obviously, there would be grave security concerns if a client were able to roam around your server's hard drives. So when you want to make the contents of a certain directory available via your Web site, you need to configure this directory either as a home directory or as a virtual directory for the site. Because you can have only one home directory per site, all other locations must be created as virtual directories.

Virtual directory names do not have to be the same as the physical directory names that they publish. For example, rather than have your client browsers use the entire path *d:\InetPub\Support\Drivers\NtDrivers\Instructions*, you can use an alias such as */NtDrvInst* to refer to the lengthy physical directory name. Once the virtual directory is created, it can be accessed via the URL *http://machine/NtDrvInst*. A virtual directory usually contains a number of files and possibly subdirectories, and it has common properties that apply to all files residing within it. However, it is possible to override the common settings for individual files in a file's Properties window. Also, if you have subdirectories within the physical location of a virtual directory, you can configure common options for that particular subdirectory that override the virtual directory's options. And if you want to go the next step, any files within the subdirectory can override the common subdirectory options. All of these overrides are accomplished using the properties settings for the specific file or directory in question. Thus, IIS provides endless possibilities for configuration options in terms of inheritance and granularity.

Application

Files within a virtual directory don't have to be grouped together for any logical reason. The virtual directory is merely a common place to store files, much as physical directories do on your hard drive. With the general paradigm shift to Internet applications, the Web server needs to have another component to describe

applications other than just a common file location. IIS uses another logical unit called an application. An application is a logical unit that potentially spawns several different virtual directories that can contain scripts, pictures, HTML files, and other files. All files and directories within an application are logically connected with each other—they are all considered part of the same application, such as an online shopping catalog, for example. You might find that the idea of grouping individual files all used for a single activity under the umbrella of "application" to be an obvious one. Technically, however, this is an important concept because all scripts and ISAPI extensions of one application share the same memory space. This means that it is possible to pass a pointer to a common memory location from one component of an application to another, thereby sharing data between them.

Configuration Options

Now it's time to get to work setting and changing server configuration options. This chapter assumes you are working with the Microsoft Management Console interface for configuring IIS. Administration tasks can also be performed by using the HTML-based administration pages. As you remember from earlier in the chapter, the HTML-based administration pages are an optional IIS component. To use the administration pages, you need access to an Administration Web site by specifying this URL:
http://localhost/iisadmin. Using the HTML-based administration pages, you can create your own scripts and applications to perform management tasks. (This will be discussed in later chapters.)

As shown in the Management Console in Figure 4.3, the server has two different Web sites: the Default Web Site and the Administration Web Site. The default Web site includes three applications: IISAMPLES, IISADMIN, and IISHELP. The default site also includes two virtual directories: SCRIPTS and IISADMPWD. Notice that the icon that represents an application looks like a little package (we will talk more about packages once we start talking about Microsoft Transaction Server), and the icon for a virtual directory looks like a folder with a little globe in the corner. Also notice that the icon associated with the Web sites is different from the one next to the Default FTP site. Icons always provide a visual clue for users.

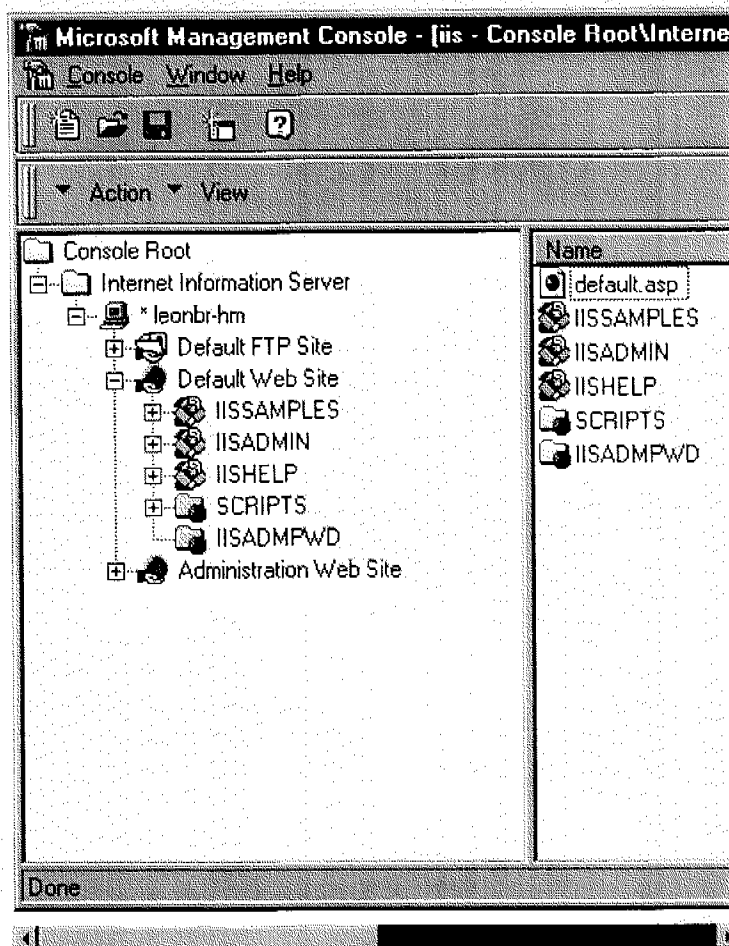


Figure 4.3 The Microsoft Management Console.

As mentioned earlier, IIS uses a hierarchical architecture. Web site objects are located one level below computer objects (indicated by the little computer icon labeled *leonbr-hm* in Figure 4.3). Virtual directories reside a level below each site. Objects on the lower levels inherit configuration parameters from the objects above, so, for example, setting values once for a computer object property makes all lower level objects inherit the same property value. This property inheritance can save a lot of time and memory storage space.

Because of property inheritance, the configuration property pages for the IIS master properties look similar to the configuration property pages for individual Web sites. If you right-click on the computer icon labeled *leonbr-hm* and choose Properties from the pop-up menu, the Properties dialog appears, as shown in Figure 4.4.

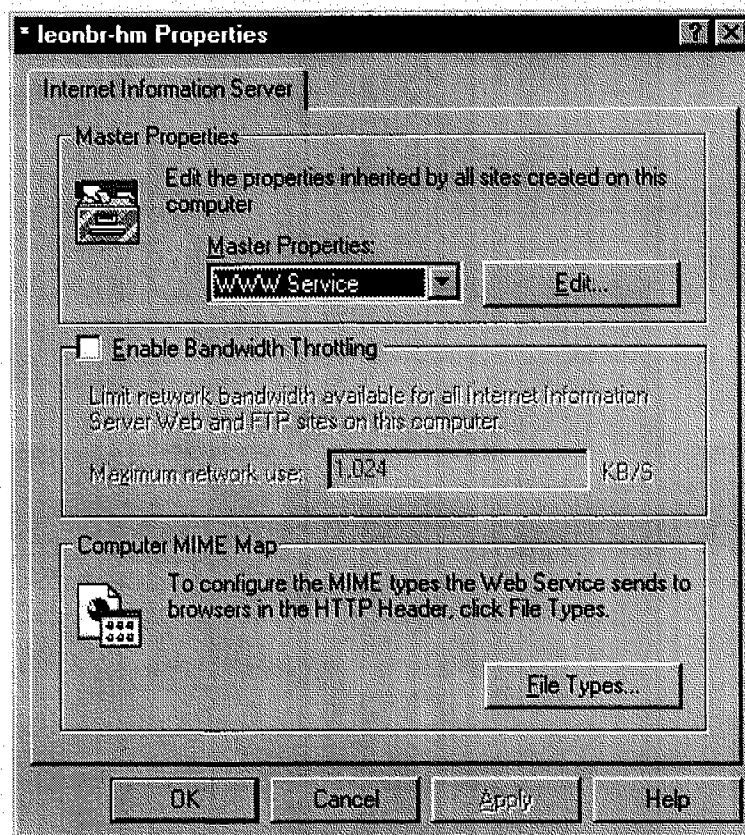


Figure 4.4 The Properties dialog box for the master computer named *leonbr-hm*.

Note Right-clicking on the computer icon also makes available the Backup/Restore Configuration option. If you choose this option, you can create a backup of the entire current configuration (WWW and FTP server settings, virtual directories, and so on). You can use this backup copy to restore configuration settings if necessary.

Master Properties Window and Default Web Site Properties Window

From the Master Properties list box in the Properties window, choose WWW Service and then click Edit to see the WWW Service Master Properties For Leonbr-hm window, shown in Figure 4.5.

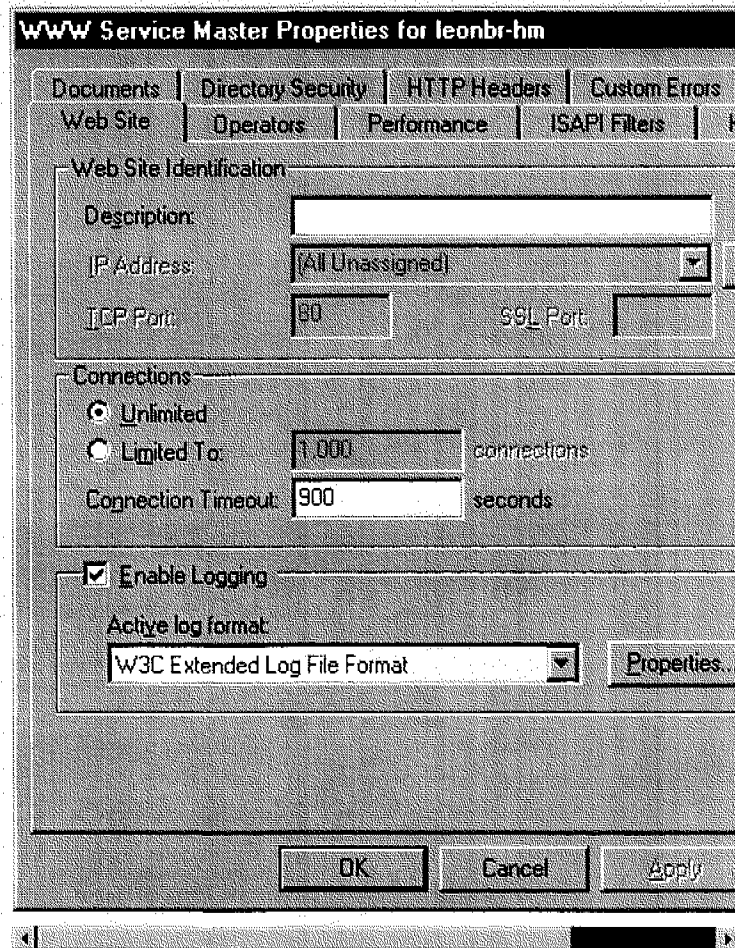


Figure 4.5 The WWW Service Master Properties window.

This window provides access to 10 different tabbed pages full of configurable property options. Similarly, if you look at the Properties window for the Default Web Site (see Figure 4.6), you will find all but one of the same tabbed pages.

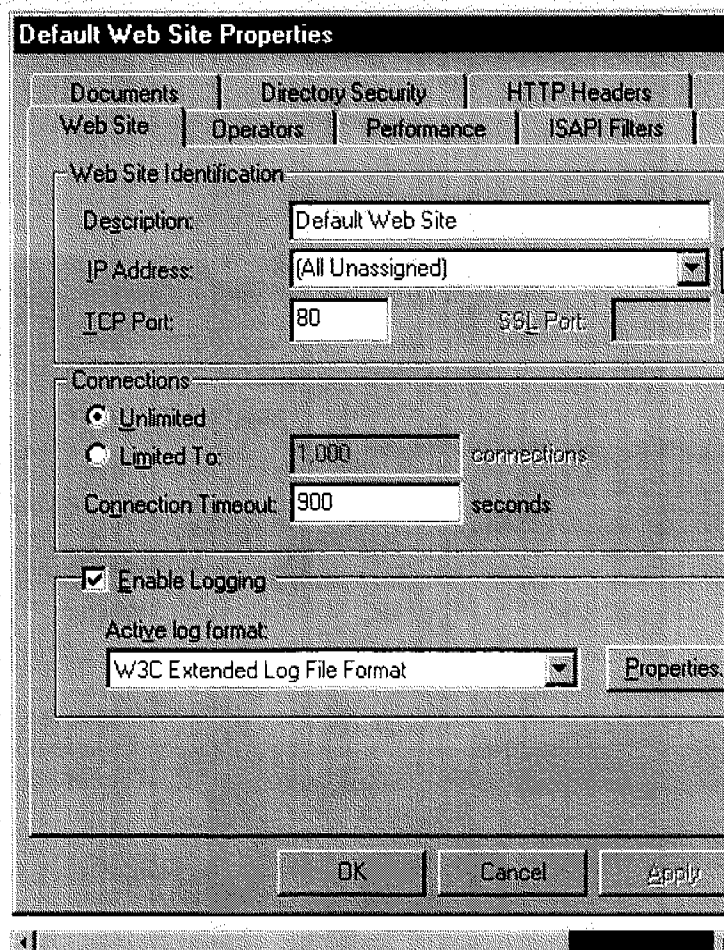


Figure 4.6 The Default Web Site Properties window.

Some property pages are available only at particular levels in the Management Console hierarchy, and the IIS 3.0 Admin page is not available at the Web Site level.

Note To see the individual configuration site settings shown in Figure 4.6 for yourself, go to the Management Console window, right-click on the Default Web Site option, and choose Properties from the pop-up menu.

The next section explains the IIS 3.0 Admin page that is accessible from the Master Properties window. Then the rest of this chapter is devoted to all the tabbed pages available in the Default Web Site Properties window.

The IIS 3.0 Admin Page

This page, shown in Figure 4.7, allows you to control compatibility with IIS versions 3.0 and earlier. Version 3.0, like 4.0, shipped with an administration tool that allowed connections to remote servers. Older versions of IIS did not support multiple sites, so an old version's administration tool does not know how to handle multiple sites on a single machine. It is still possible to manage IIS 4.0 with the tool shipped with versions 3.0 and earlier, but only one site can be managed per machine. This page indicates which site it will be. (A similar option for FTP services will be discussed in Chapter 9.)

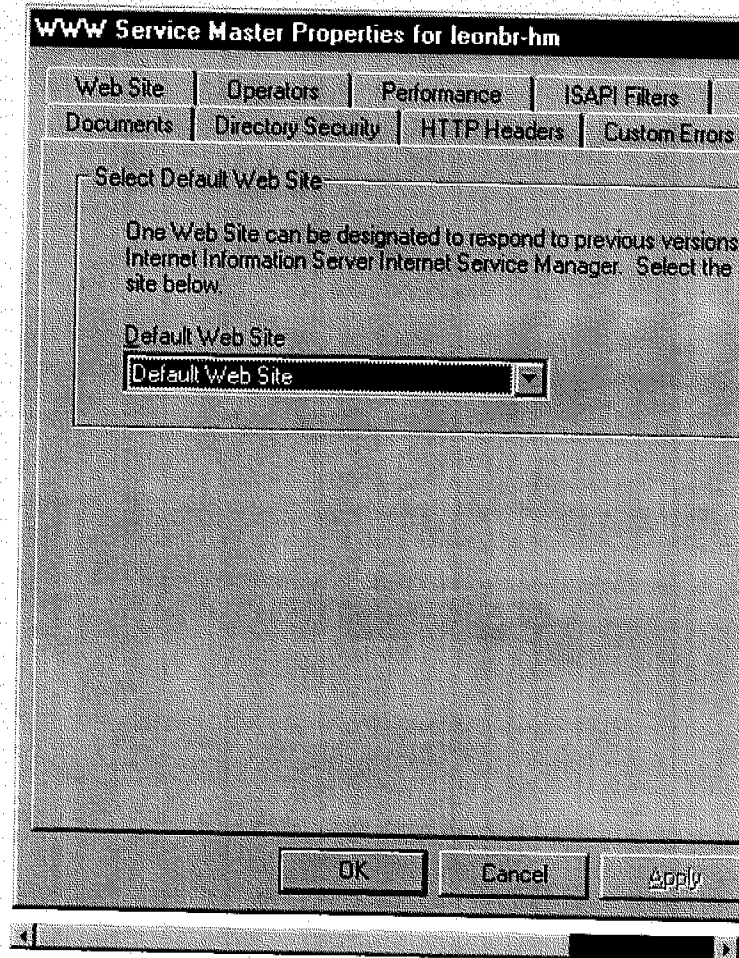


Figure 4.7 The IIS 3.0 Admin page.

The Web Site Page

As shown in Figure 4.6, you can use the Web Site properties page to change a description of the site as well as to change the default port (port 80) to a different port. If you change the port from the default, however, your clients will need to know the new port number to include in the site's URL. For example, if you change the port to 4000, the URL might look something like this: *http://machine_name:4000/default.htm*. Why would someone change the default port number? One reason the site administrator would do this is to make the site available only to users who know the port number assigned to that site, thus limiting the site to a specific group of users.

To associate more than one address or a different port to the same Web site, you can click the Advanced button on the right of the IP Address text box in the Default Web Site Properties window. This brings up the Advanced Multiple Web Site Configuration dialog box shown in Figure 4.8.

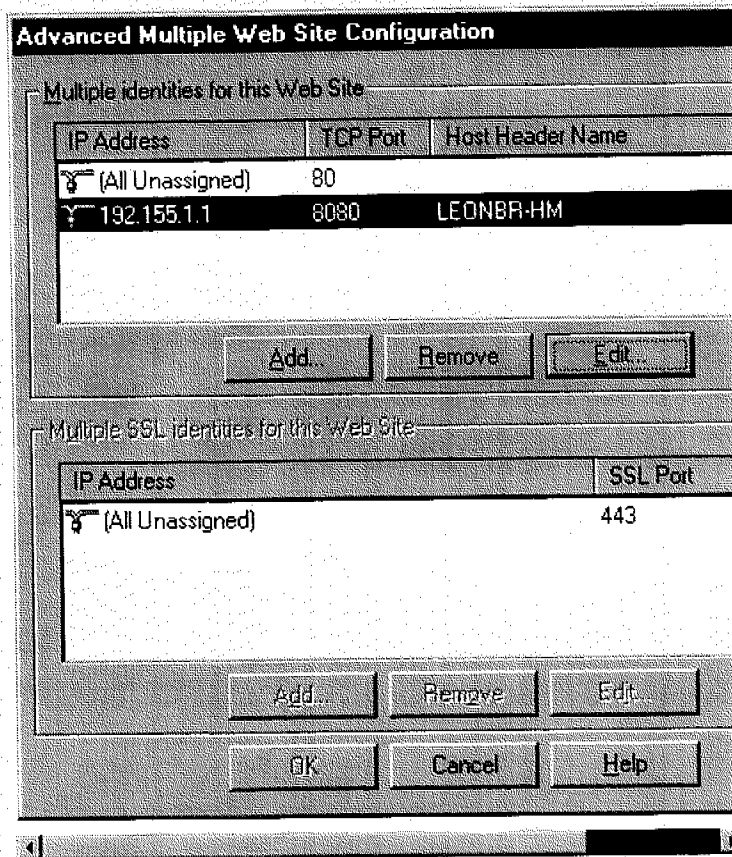


Figure 4.8 Using the Advanced Multiple Web Site Configuration dialog box to associate more than one IP address and port with the server.

Because it is possible to assign more than one computer name to the same IP address (by adding entries to your DNS server or by editing your TCP/IP HOSTS file), the server needs to have a unique identifier for each different site. Clients compatible with HTTP version 1.1, such as Microsoft Internet Explorer 4.0, support the Host header that helps differentiate requests to multiple computer names. Headers are a way for a server and browser to exchange auxiliary information. (Chapter 22 will cover headers in more detail.)

For instance, suppose you have two host names, *machine1.domain.com* and *machine2.domain.com*, which both resolve to the same IP address, *123.123.123.123*. The server administrator will probably want to use different sites for each machine name. Because both names will eventually be resolved to the same IP address (*123.123.123.123*), the browser needs to pass information about what host name was specified in the requested URL. That is precisely what the Host header does. It reflects the target host of the request. So a request to *http://machine1.domain.com* will have this Host header:

```
Host: machine1.domain.com
```

After the request has arrived at the server, the Host header is examined and the correct Web site is determined. If for some reason the requested site is not available or the Host header is missing, the default page from the default site is sent.

Other configuration options accessible from the Default Web Site Properties window include restrictions on the number of simultaneous connections allowed to the server and the connection timeout period for idle sessions. By default, IIS uses HTTP version 1.1. This version of HTTP has a provision that allows connections to stay open (that is, to not close the underlying socket) even after the request has been completed, which can greatly improve performance if you are making multiple requests to the same server. When no requests occur after the specified timeout, the server closes the socket associated with that session.

Glossary

socket Communication endpoint used in low-level network protocols (such as TCP/IP). Clients and servers use sockets to exchange information remotely.

You can also specify options for logging server activity on the Web Site property page. (You can read about logging options in detail in Chapter 6.)

The Operators Page

The Operators property page shown in Figure 4.9 is fairly simple. Individual users or groups listed here as operators have the right to administer the Web site. Administration can occur either through the Management Console, the HTML-based administration pages, or the management programming interfaces. This page simply displays the users and groups that have operator privileges to write changes to the metabase, where configuration information is stored.

Glossary

metabase A special storage area used by IIS to store its configuration settings.

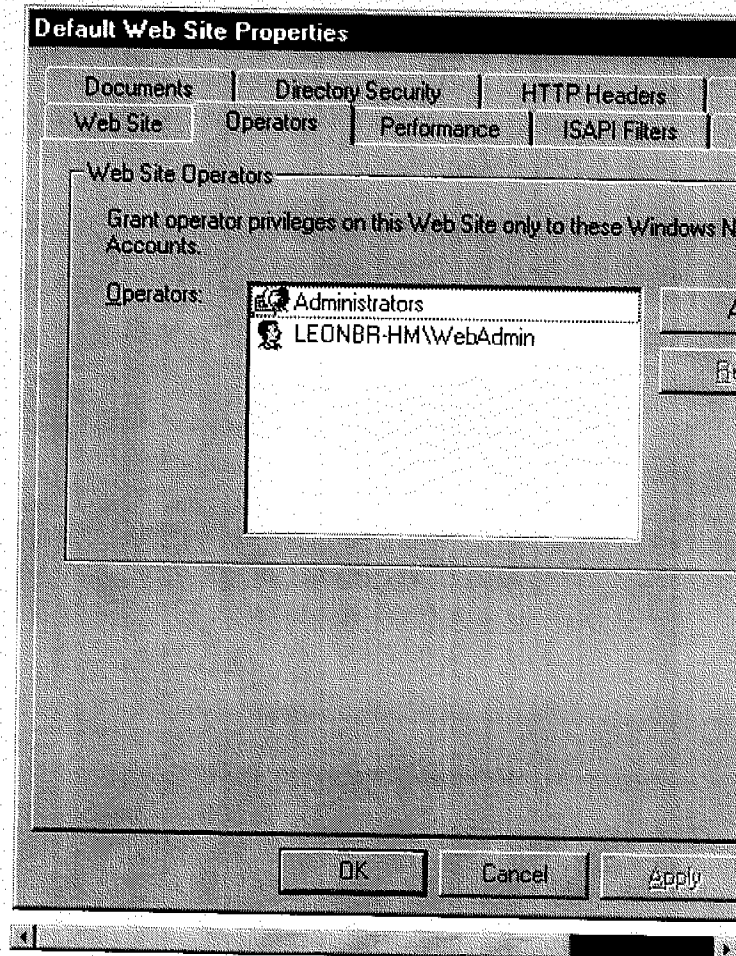


Figure 4.9 The Operators page in the Default Web Site Properties window.

The Performance Page

The Performance page, shown in Figure 4.10, allows you to optimize performance of the server based on the number of anticipated requests per day. This has the effect of determining how much memory IIS will reserve for caching request information. If your server receives a high number of requests, more of the server's memory will be used by IIS's cache. In the case of fewer requests, setting this option too high will waste memory by saving unnecessary cache space.

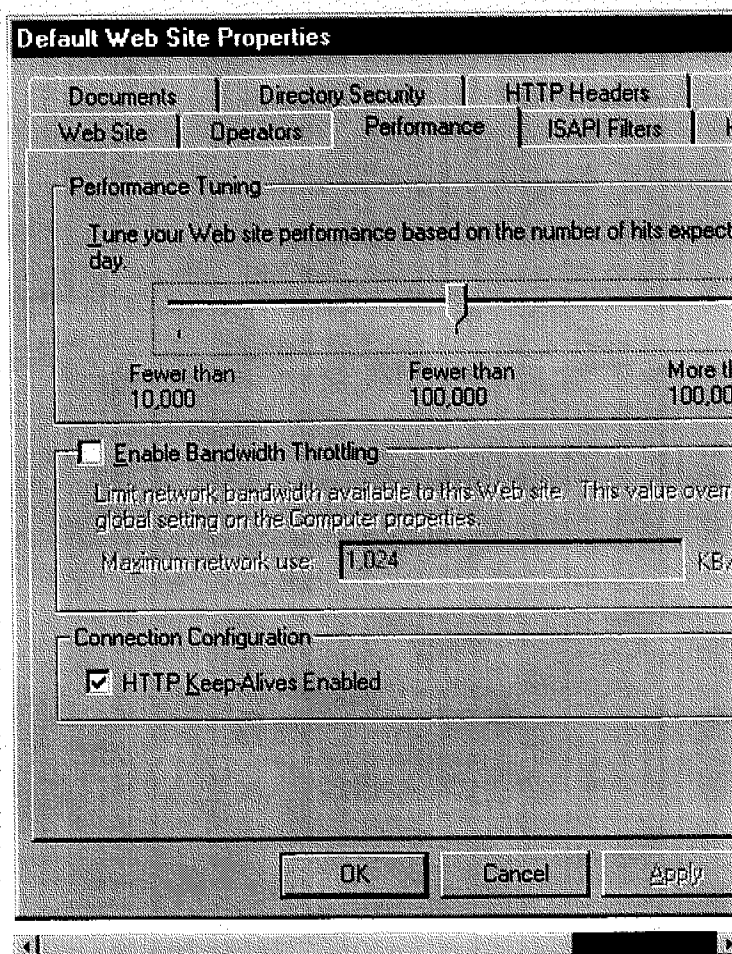


Figure 4.10 The Performance page in the Default Web Site Properties window.

The Enable Bandwidth Throttling option allows you to limit the network bandwidth used by this specific site. Site bandwidth settings take precedence over the master WWW settings.

In the Connection Configuration section there is an option to enable or disable HTTP Keep-Alives. Remember our note on keeping sockets open to improve efficiency? Checking this box (the default) indicates that Keep-Alives are enabled.

The ISAPI Filters Page

The ISAPI Filters page is shown in Figure 4.11. ISAPI filters are covered in detail in Chapter 21, but for now you can think of them as special DLLs that are loaded by the server and are notified at particular points for each request received by the site. This page shows ISAPI filters listed for this particular site (namely, SmartRedir or the Smart Redirector filter). In general the Web server might have global ISAPI filters that are installed through the Master Properties window. Global ISAPI filters receive notifications for all Web sites on the machine. Even though Figure 4.11 does not list global scope ISAPI filters, they still affect all sites because of the inheritance architecture of IIS objects.

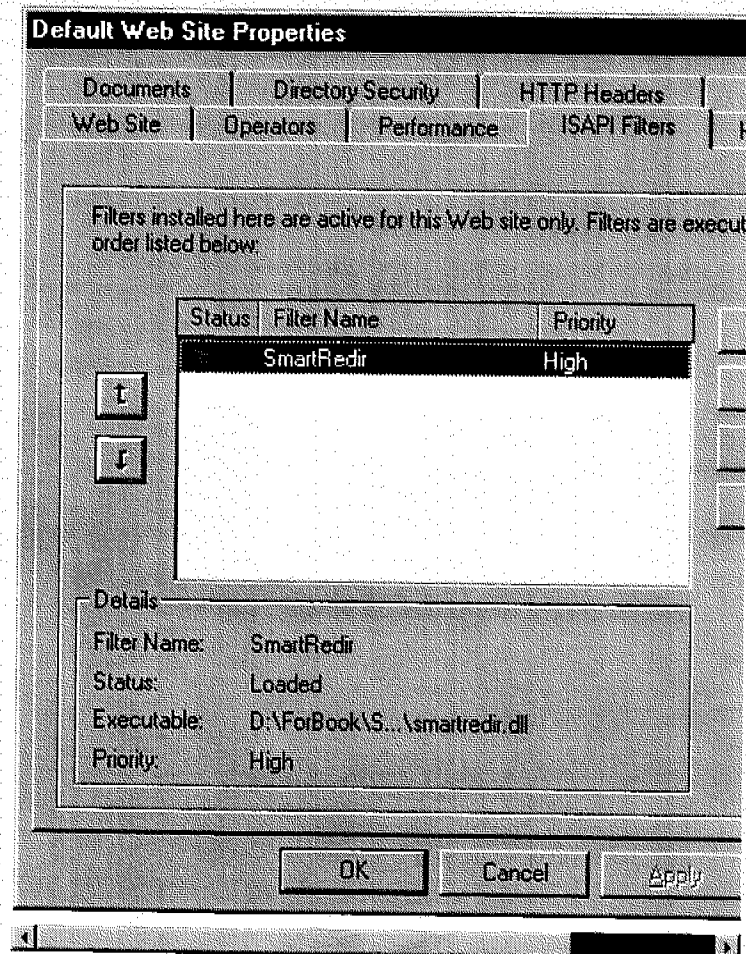


Figure 4.11 The ISAPI Filters page in the Default Web Site Properties window.

After a filter has been loaded by the server, the internal filter properties are shown on the property page. The filter listed on this page controls which browsers are allowed to access certain directories. We will look into this filter in greater detail in Chapter 21.

If more than one filter were loaded, you could arrange for them to be called in a specific order to allow incoming requests to be handled by multiple filters in that order. The ISAPI Filters page also shows you at a glance the specific priorities for any loaded filters. This can be a handy feature when you're trying to track down ISAPI filter problems.

The Home Directory Page

The Home Directory page is shown in Figure 4.12. This page contains items similar to those you would see for any virtual directory.

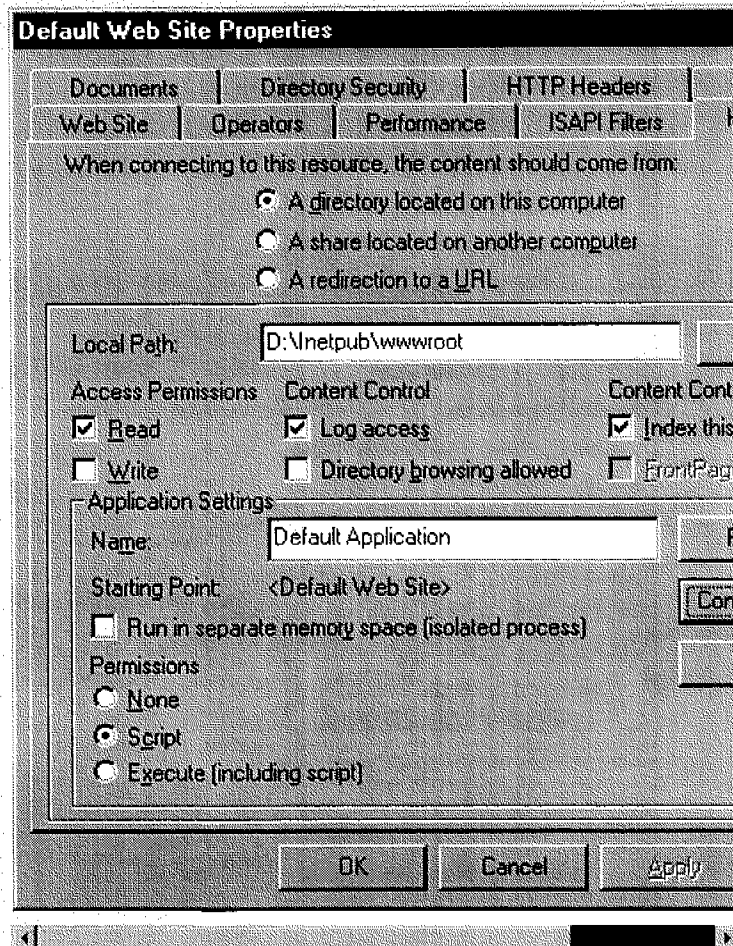


Figure 4.12 The Home Directory page in the Default Web Site Properties window.

Share location You can change many Web site options from this page, including the physical location where the Web content is stored. In the simplest scenario, the home directory resides on the local machine, as shown in Figure 4.12. It is possible, however, to make IIS serve content from other machines on the network. In that case, the machine that contains the data does not necessarily have to run IIS. The machine is addressed via the UNC (Uniform Naming Convention) in this way: \\server\share. Because the IIS process itself runs under the Local System account, the account is known only to the local machine and cannot connect to remote shares on the network. (The mysteries of the Local System account—that is, the account built into the Windows NT operating system—will be explained in Chapter 12.) IIS avoids this problem by prompting for the user name and password that are necessary to connect to a remote machine. To enable this feature, from the Home Directory page choose A Share Located On Another Computer. When you do this, the Connect As button will appear. Click it to enter the credentials (user name and password) required to connect to the remote share.

Glossary

Local System account An account built into the Windows NT operating system. This account is used internally by Windows NT to run system services, such as the IIS Web Publishing Service.

URL redirection IIS can not only serve content from a remote machine, it can also initiate redirection. For instance, assume the server is configured to provide content for a virtual directory from the /my_org_content directory. But for some reason this directory needs to be changed. This could be necessary because of changes in the drive's partitions or in NTFS permissions, directory structure changes, the directory might need to be renamed, or some other reason. You could create new virtual directories with a new name and create links to the new location. This will, of course, affect bookmarks made earlier to the original location. Instead of allowing the browser to report a "URL Not Found" error to the user, you can configure the server to send a new location to the browser when a request is made for the original location. This is called a *server redirect*. IIS sends a special HTTP status code to the browser with the new URL; if the browser supports redirection, it sends a request to the new URL. When you select the option labeled A Redirection To A URL, the Home Directory page changes to look like the one shown in Figure 4.13.

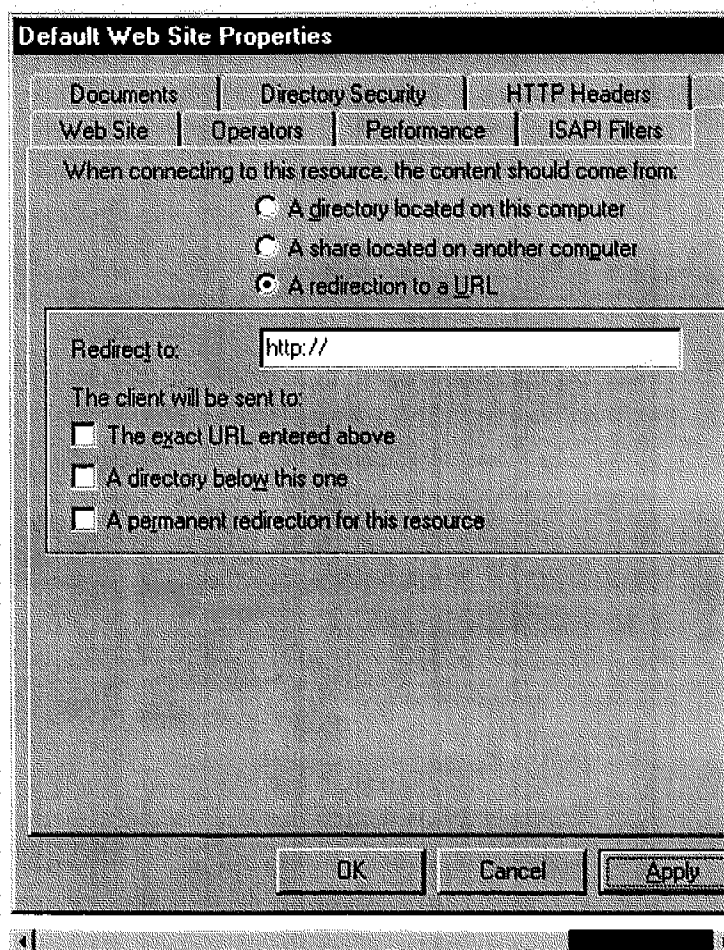


Figure 4.13 The Home Directory page in the Default Web Site Properties window when Redirection is selected.

You can choose from three different redirection options, as shown in the lower half of the Home Directory page. By checking The Exact URL Entered Above, you can redirect all requests made to any element in the virtual directory to a single file. For example, if you have a virtual directory named /my_org_content, a request to `http://myserver/my_org_content` or to any file in this directory (such as `http://myserver/my_org_content/myfile.htm`) could be

mapped to a single HTML file, such as *MyExplanations.htm*. This option can be helpful when the content of the original directory is not available and you want to explain why. The Redirect To text box contains the URL of the explanation file that will be returned to the client.

The next check box in this section, *A Directory Below This One*, forces redirection to a child directory. For example, entering *mysubdir* as a redirect target for a virtual directory named */support* will force all requests to *http://myserver/support* to be redirected to *http://myserver/support/mysubdir*.

When server redirection occurs, IIS sends a "302 Temporary Redirect" HTTP status code to the client along with the new location. Checking the last option, *A Permanent Redirection For This Resource*, forces the server to return a "301 Permanent Redirect" HTTP status code to the client. Upon getting the 301 code, a smart browser can even update saved bookmarks.

Server variables and wildcards If you think that IIS's redirection is limited to simple redirection from place to place, you should know that its real power comes from the ability to handle server variables and wildcards. Server redirection variables are special characters inserted in the "Redirect to:" URL. These variables allow you to transfer part of the original URL to the new location. The following table shows the redirection variables and how they are used.

Variable	Function	Example
\$S	Passes the matched suffix of the original URL to the target URL	For <i>http://server/dir/doit.bat</i> , \$S represents <i>/doit.bat</i> . If the suffix does not exist (<i>http://server/dir</i>), \$S is blank.
\$P	Passes parameters from the original URL	For <i>http://server/dir/doit.bat?P1=1</i> , \$P represents <i>P1=1</i> .
\$Q	The same as \$P, but with the question mark	For <i>http://server/dir/doit.bat?P1=1</i> , \$Q represents <i>?P1=1</i> .
\$V	Deletes the server name from the original request	For <i>http://server/dir/doit.bat?P1=1</i> , \$V represents <i>/dir/doit.bat</i> .

Here's an example to help illustrate how server variables work. Suppose that we have configured all requests to the virtual directory */support* to be redirected to the exact URL */target\$S*\$P*\$Q*\$V*. Notice here that requests are being redirected to the virtual directory */target* along with all server variables, which are separated by asterisks to allow an easy match of the resulting substitutions with corresponding parameters. As soon as we enter *http://server/support/doit.bat?Param1=1* in the address line of the browser, IIS redirects the request to the following location: *http://server/target/doit.bat*Param1=1*?Param1=1*/support/doit.bat*

In the above sample URL, \$S is matched with /doit.bat. As you can see "/" is included. Therefore, we don't need to precede \$S in the target URL (/target\$S*\$P*\$Q*\$V) with "/". The \$P is substituted with Param1=1 and \$Q with ?Param1=1. The last variable, \$V, has generated the /support/doit.bat string by omitting the server name.

Server variables are powerful (and difficult to use), but wildcards are even more powerful. Wildcards allow you to match any number of characters in the original URL to a pattern in the destination URL. The asterisk (*) wildcard character denotes the portion of the URL that can take any combination of characters. There are, however, some rules to follow when you're using wildcards. You must make sure that The Exact URL Entered Above option is checked in the Home Directory page. Also, the target URL must start with the wildcard character (*), and different pairs of wildcards must be separated with a semicolon.

For example, to redirect all requests for *.ASP files to my_new_default.asp and all requests for *.bat files to all_in_one.bat, type this for the "Redirect to" URL: *.*.asp;/my_new_default.asp;*.bat;/all_in_one.bat. You can get the matched wildcard part by using the special variables \$0 through \$9. Now all *.asp requests are redirected to the my_new_default.asp file. What would happen if a browser sent a request for the my_new_default.asp file itself? If you don't let IIS know that this file is an exception to the virtual directory redirection order, an infinite loop can occur. By using the exclamation mark (!) as a redirect URL for the my_new_default.asp file, you let IIS know that this file is an exception to the redirection. The "!" needs to be entered as a redirection target for my_new_default.asp file. To do this, right click on the individual file (my_new_default.asp) and set its individual redirection property. This is one of those times when the ability to set properties for a specific file comes in handy.

Access permissions Setting and controlling access permissions is an important part of running IIS. When redirection is not selected, the Home Directory page allows you to control read and write permissions for a directory. Read permission is self-explanatory—when you grant permission to read, anyone requesting that file is allowed to read it. Write permission, however, is a little more complex. HTTP version 1.1 provides a *PUT* method that allows files, such as HTML documents, to be uploaded directly to the Web server. Checking the Write access permission will allow browsers to upload documents and files to this virtual directory. This right is also essential to allow posting files with the File Upload control, which is discussed in Chapter 8. If the actual directory is located on an NTFS volume, IIS will also check the NTFS permissions before reading or writing, and the most restrictive permissions will take precedence. (NTFS is covered in Chapter 12.)

Content control The four Content Control options—Log Access, Directory Browsing Allowed, Index This Directory, and FrontPage Web—are found near the middle of the Home Directory page when redirection is not selected.

IIS can generate entries in the log file whenever particular resources are accessed. You can enable or disable this for particular sites, virtual directories, directories, or individual files by checking the Log Access check box. IIS allows you complete control over resource logging capabilities.

Another feature of IIS is its ability to perform directory browsing. If a directory contains no default document (usually DEFAULT.HTM or DEFAULT.ASP) and the browser does not specify an exact filename, the server can send back an HTML-formatted directory listing. It will look similar to what your browser displays when it accesses an FTP server. Because looking at the raw directory structure of the server can be somewhat revealing to the user, you should exercise caution before allowing directory browsing. Directory browsing is disabled by default.

Microsoft Index Server is an integral part of IIS, and you can enable its power by checking the Index This Directory check box. Chapter 7 will take an in-depth look at Index Server, but here you need to know that Index Server provides a search engine for finding content on your server. It performs this feat by indexing the specified contents on the server and keeping the results in a local database of keywords and properties. This allows users to query Index Server for any combination of words to see a list of links to specific files that match the search criterion. A background process in Index Server monitors changes in virtual directories and updates the index database accordingly. If the Index This Directory option is not checked, Index Server will neither create nor update the database with information about the contents in this directory.

If the FrontPage server extensions are installed, checking the FrontPage Web check box makes this directory available for access by FrontPage server extensions. When checked, this option allows a user with the appropriate permissions to edit the contents of this directory using the FrontPage HTML editor.

Application settings As mentioned earlier, an application is a logical unit that can tie together many different subdirectories and individual files into one logical piece. ISAPI extensions, ASP scripts, and server-side components all share a single object context and memory space that is configured from the home directory or virtual directory level. If you want, you can convert a virtual directory to an application. Figure 4.12 on page 69 shows a virtual directory that has already been converted to an application named Default Application.

In the bottom section of the Home Directory page, a Create button will appear where the Remove button is located (in the figure) if the virtual directory is not an application. To create an application, click the Create button and enter an application name in the Name text box. After you do this, the icon in the Management Console will change from a folder (that is, a directory) to a package (that is, an application).

Application configuration Many configuration options are specific to particular applications. Clicking the Configuration button in the Home Directory page opens the Application Configuration window, which accesses all of the applications. Remember that properties are inheritable, so all options for a higher-level application (such as the default Web page) are inherited by lower level applications.

The App Mappings page in the Application Configuration window, shown in Figure 4.14, allows you to map specific file extensions to ISAPI extension DLLs that will be executed when requests for files with those extensions are received. For instance, the .ASP file extension is mapped to the ISAPI extension ASP.DLL. Thus, when a request is received in this application for a filename with an .ASP extension, ASP.DLL is called to handle the request. You can modify

the application mappings (with potentially disastrous results), or you can create application mappings for your own files. You can even add to the extensions handled by existing applications. For example, you might have some ASP files with the extension .FOO. You can use this page to add a mapping to associate the .FOO extension with ASP.DLL.

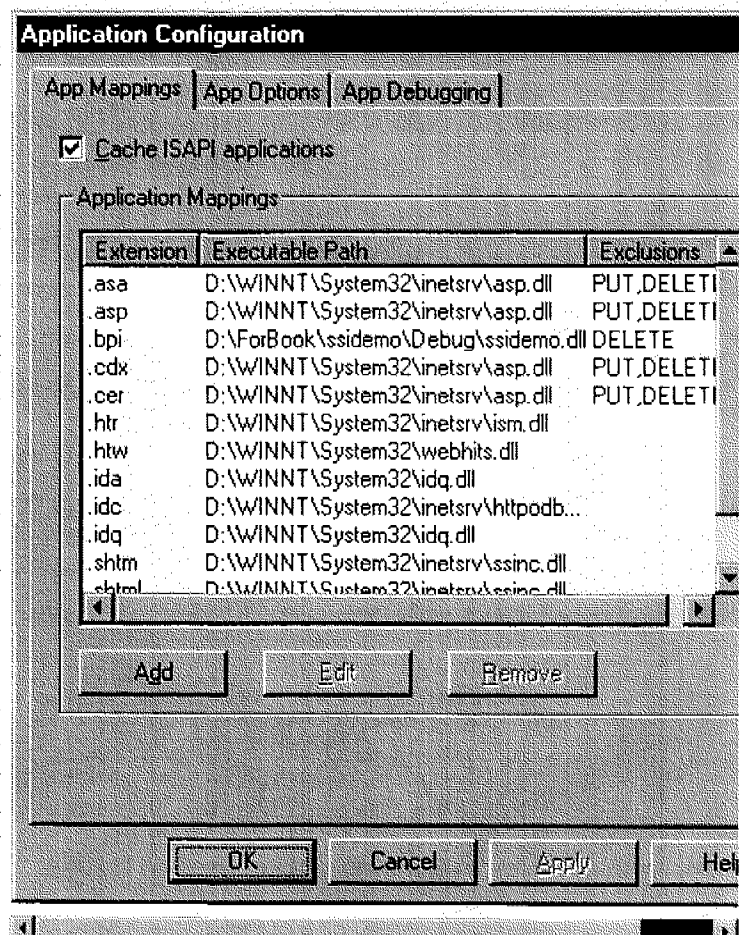


Figure 4.14 The App Mappings page in the Application Configuration window.

Note In Chapter 21 you will learn that you can write your own DLL to allow custom processing of specific file types. For example, we wrote a sample ISAPI extension DLL named SSIDEMO.DLL that handles files with the .BPI extension. (BPI stands for "Braginski Powell Include"—pretty creative, eh?) The application mapping replaces all tags (such as *AUTHOR_NAME*) in *.BPI files with a predefined value from a registered dictionary file. The dictionary file contains associations such as *AUTHOR_NAME equals "Leon Braginski"*. The result is that files with the .BPI extension can contain templates, instead of real text, that will be sent to the browser as a perfect HTML with the tags appropriately replaced.

You can click the Add button in the App Mappings tab to open the dialog box shown in Figure 4.15 to add *.BPI files to IIS's extension mappings. In this dialog box, you enter the name of the executable or DLL that will be processing the designated file types. In this case, the path to SSIDEMO.DLL is listed, with specifications that it handle files with the .BPI extension.

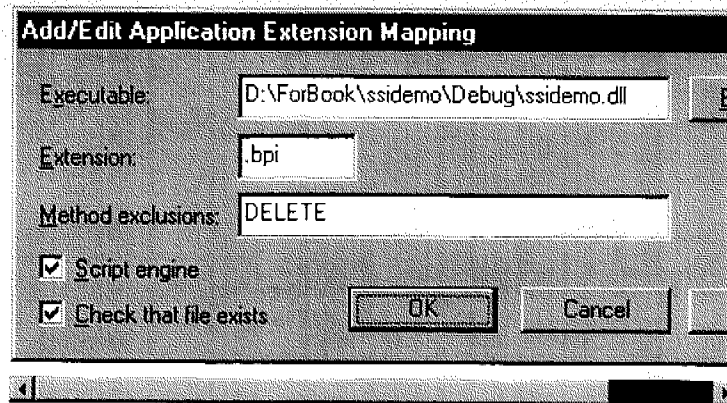


Figure 4.15 The Add/Edit Application Extension Mapping dialog box.

You can avoid invoking SSIDEMO.DLL even when a request is made for a *.BPI document by creating a list of exclusion methods. Exclusion methods are the HTTP verbs or methods for which you do not want IIS to execute your extension. More than one method can be entered, and each must be separated by commas. SSIDEMO.DLL is a simple extension and can't handle the DELETE HTTP verb; therefore, DELETE has been added to the exclusions list in the dialog box.

There are two more important check boxes in this dialog box: Script Engine and Check That File Exists. Normally, an EXE or a DLL can be invoked only from a virtual directory that has execute permissions. Because of this, .BPI files would normally need to be located in a directory that has execute permissions. But this problem can be avoided by checking the Script Engine option, which makes file invocation possible from any directory, even without execute permissions. The second option allows you to check for the existence of a script file before invoking an application to process it. This option lets you take advantage of IIS's ability to generate errors for files that do not exist. It will also catch situations in which the client does not have permissions to read a given file.

Back at the top of the App Mappings property page is another configuration option: Cache ISAPI Applications. One of the performance enhancements that ISAPI applications have over CGI applications is that ISAPI applications can be loaded into memory once and reused to handle subsequent requests. This avoids the overhead of having to reload an application into memory repeatedly. Keeping an ISAPI extension loaded in memory does pose a problem, however, particularly when it is in its development stage. If the DLL is cached in memory, newer versions of the DLL cannot be copied over the old one because it is considered to be "in use." Removing the checkmark from this option allows IIS to immediately unload the DLL once a request is finished. Of course unchecking this option can seriously affect the performance of IIS and ISAPI extensions, but it can be a useful tool for development.

The next page in the Application Configuration window, App Options, is shown in Figure 4.16. This page deals mainly with the configuration of ASP applications. In Chapter 17, these configuration concepts are discussed in detail, so the following is simply a quick description of each.

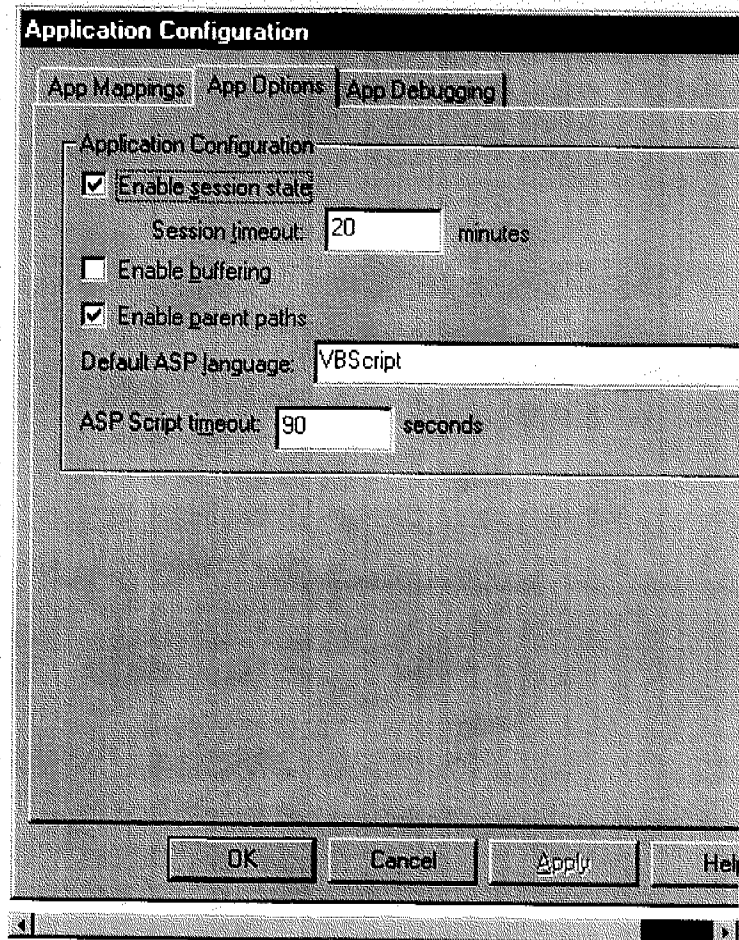


Figure 4.16 The App Options tab in the Application Configuration window.

For each user who accesses an ASP page, the server can create a session object. Session objects allow a user to be identified across multiple page requests and provide a means for preserving information across those requests. Because using ASP session objects requires some overhead, and because ASP requires the use of HTTP cookies to preserve state across requests, developers might want to avoid using session objects. The Enable Session State check box lets you turn on and off the underlying support for session objects for the particular application. If sessions are enabled, the time-out value equals the length of time a user's session object will be held in memory without receiving subsequent requests before it is destroyed.

Checking the Enable Buffering check box tells ASP to collect the entire script's response before sending it to the client. Otherwise, the server sends the script output immediately to the client as it interprets the script. An advantage to enabling buffering allows a script to set HTTP headers anywhere within a script since the HTTP headers won't be sent until the entire script has finished executing. If buffering is not enabled, all HTTP header manipulation must take place before a single byte has been sent in the response body.

Checking the Enable Parent Paths option allows scripts to access files in parent directories using the double-dot (..) notation (for instance something like `..\scripts\fdisk_server_drive.asp`). If you enable parent directories, you should not set execute permission

on them because it can provide a means for a script to execute an unauthorized program.

ASP scripts can use any number of script languages, although IIS installs with support for only VBScript and JScript. You can, however, write your own script interpreter or get a third-party product (such as REXX or Perl). If a default ASP language is indicated in the Default ASP Language box, any script in an ASP file is considered to be written in that language unless otherwise specified. Specific ASP files can override this option by including the `<%@ LANGUAGE %>` directive inside the ASP file.

You can also specify an ASP script timeout value in the appropriate box, which will cancel the execution of any ASP scripts that have not completed by the specified time. When the timeout period has expired, the running script is terminated and an event is written to the Windows NT event log. This can help avoid problems with script files that have infinite loops or that are perpetually blocked waiting for a troublesome server component. However, too short a timeout specification can create a problem if legitimate scripts take a long time to execute. You can make the timeout indefinite by setting this value to `-1`.

Finally, the App Debugging page of the application Configuration window, shown in Figure 4.17, allows you to configure various ASP debugging options. If script debugging is allowed, upon encountering an error ASP starts a debugger that helps find the problem. Client debugging is not implemented at this point and is reserved for future use.

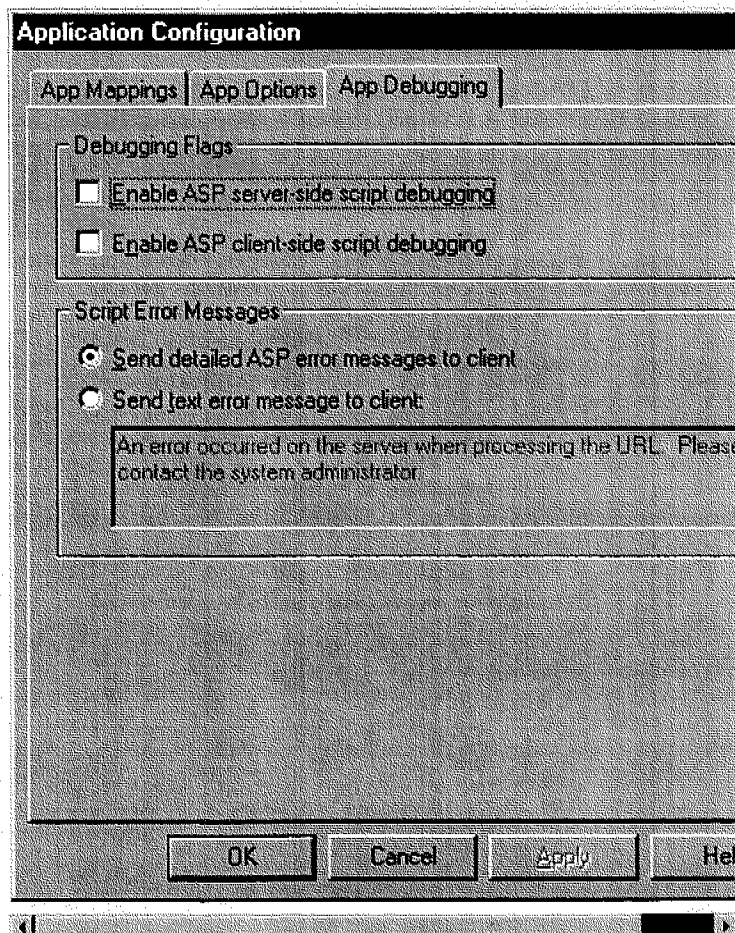


Figure 4.17 The App Debugging page in the Application Configuration window.

The Script Error Messages options allow you to send either a specific ASP error that points out the details of the error in the ASP script or a simple static error message. Getting details of script errors is useful when you are developing an application for the Web, but you might not want to expose the details of your ASP code to users who access your Internet site. Therefore, you can choose the Send Text Error Message To Client option to return the specified text for all script errors encountered by this application.

Other options on the home directory page Take another look at Figure 4.12 on page 69. The parameters in the Application Settings area affect how IIS handles scripts and ISAPI extensions. The Run In Separate Memory Space option forces IIS to load all components of the application in a memory space separate from the IIS process. Normally, ISAPI extensions are loaded in the main IIS process space. A poorly written or malicious ISAPI extension could crash the entire Web server (even though IIS tries to prevent this by using such programming techniques as exception handling). When applications run in their own processes, the risk of server instability is reduced.

Precise control of which applications can and cannot be executed is an integral part of server security. You can achieve such control by setting up correct execution permissions. If you select None in the Permissions section at the bottom of the Home Directory page, no code will be able to run in this application (or in the virtual directory, depending on which property pages are opened).

Note The Home Directory page for a Web site (or the Virtual Directory page for lower level entities) looks very similar for virtual directories and for applications. Remember that you can convert any virtual directory to an application. Virtual directories that are not applications disable the Configuration button as well as the Run In Separate Memory Space check box.

Furthermore, script files such as ASP or IDC files are not simple static files but are executed by script maps. For script files to run properly, IIS needs to invoke a server-side script interpreter, such as the SSIDEMO.DLL interpreter that was configured earlier in the chapter for *.BPI files. Choosing the Script option when installing this interpreter indicates that only Script-level access permissions are required for these files to run. But how about executables such as CGI and ISAPI applications? These applications are considered the most serious risks to server security, so to permit executable files to run, you must choose the Execute option. This option will also allow scripts to run.

The Documents Page

Figure 4.18 shows the Documents property page in the Default Web Site Properties window. With the exception of the title bar, this page looks the same whether it is accessed for Web sites or for virtual directories.

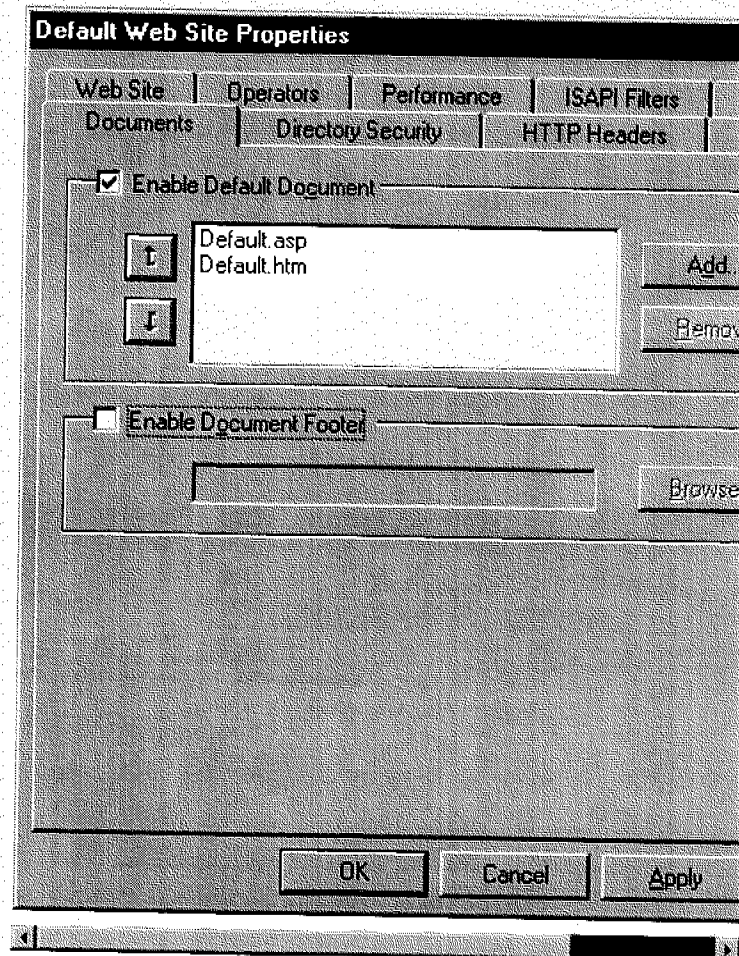


Figure 4.18 The Documents page in the Default Web Site Properties window.

As mentioned earlier, when a client sends a request to the server without specifying a filename (for example, *http://www.microsoft.com*), the server sends a default document to the client. Once you check the Enable Default Document check box, you can use the Add button to configure multiple default documents so that IIS will search for each specified document in the list until it locates one.

One of IIS's new features is its ability to add a footer file to all your responses by placing a check mark next to Enable Document Footer and specifying the filename. For example, if you have bunch of pages and want to add the same copyright statement to the end of each one, you can create a little HTML file with the indicated statement and use this document as a footer. By doing this, you can avoid the tedious job of typing the same sentence at the end of each file. Notice that the footer file is not a full-blown HTML document with typical HTML tags. It is simply a piece of HTML that is appended to normal responses.

The Directory Security Page

The Directory Security page in the Default Web Site Properties window, shown in Figure 4.19, is a gateway to many specific options related to IIS security.

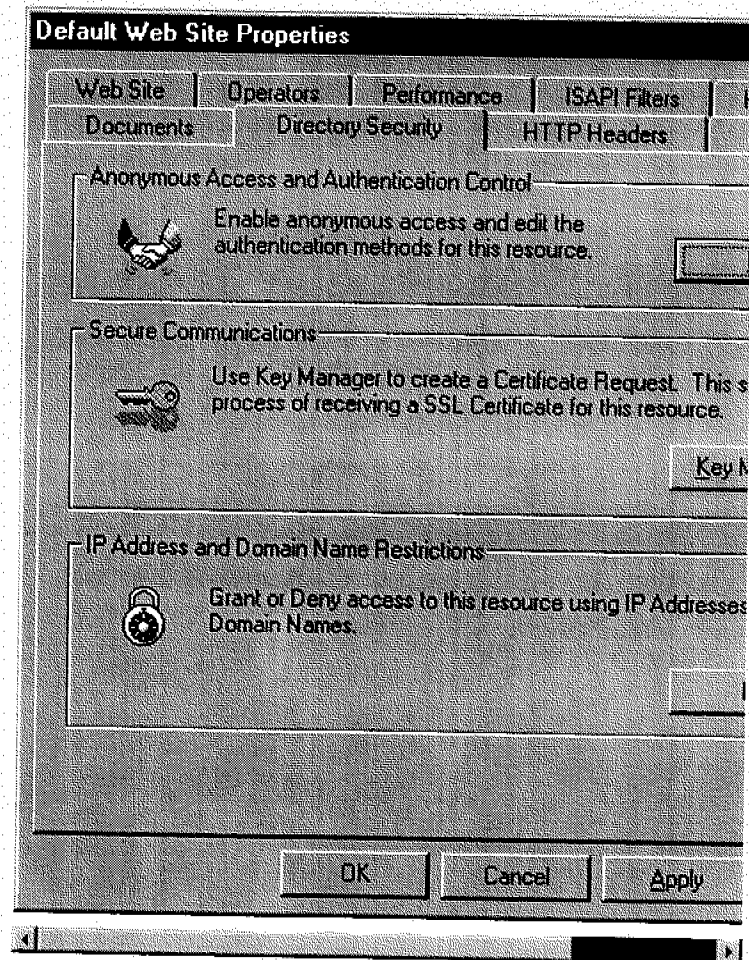


Figure 4.19 The Directory Security page in the Default Web Site Properties window.

Before you jump into the mysteries of IIS security, consider some basic security concepts. Under Windows NT, every process must run in the context of a specific user account. A user can log on to a network via either a local account or a domain account if the machine participates in a domain structure. IIS is a Windows NT service that runs as the Local System account. Each machine has its own unique Local System account with virtually unlimited rights on that machine. However, since the credentials differ for each machine's Local System account, trying to access other machines on a network using the Local System account for one particular machine will result in an access deny error. That is why, when we specified a virtual directory located on a UNC share, we had to enter a valid username and password on the target machine.

Because of the broad scope of permissions available on a local machine for the Local System account, running scripts, ISAPI DLLs, and CGI executables under this account could cause a potential security disaster. To prevent this, IIS creates a local account with limited rights. This account is usually named `IUSR_machine_name` (*machine_name* is replaced with the name of your computer) and services each anonymous connection from a browser.

For example, say a browser requests a file named `SECURE.HTM` from IIS. The server tries to access the specified file by impersonating the `IUSR_machine_name` account. If, however, the

IUSR_machine_name account doesn't have permissions to read the SECURE.HTM file, IIS will send a specific HTTP error to the client. In response to the error, the browser will ask for a username and password and submit the newly acquired credentials to the server. The low-level details for HTTP authentication are discussed in Chapter 22, but for now assume that the server has the user's credentials and will impersonate this account to try to access SECURE.HTM. Assuming the account has the appropriate privileges, the file will successfully be sent to the client.

Anonymous access and authentication control A number of different schemes govern how the username and password provided by the client's browser is authenticated by the server. IIS has built-in support for two such schemes: Basic Authentication, in which credentials are Base 64 encoded and transmitted in clear text; and NTLM (NT LAN Manager), which is a secure challenge/response authentication scheme that can be used only by Windows-based machines. NTLM is a more complex scheme: the password is never actually transmitted over the network, and credentials are verified by the valid encryption and decryption of an initial challenge.

Glossary

Base 64 encoding A special encoding scheme in which all characters use the Base 64 numbering system. Just as computers use the binary numbering system (only two digits exist: 0 and 1), a Base 64 system uses 64 different characters.

challenge/response authentication A special method of securely verifying a username and password without actually transmitting the information over the network.

Note For NTLM authentication, if the client and server participate in a domain-based security scheme, a browser might not even ask the user to enter his or her name and password. In this instance, the credentials of the currently logged-on user are utilized to perform the authentication.

IIS can be configured to use any combination of NTLM, Basic, and Anonymous authentications. If you click the Edit button in the Anonymous Access section of the Directory Security page, the Authentication Methods dialog shown in Figure 4.20 appears. From this dialog box, you can edit parameters for anonymous access and authentication control.

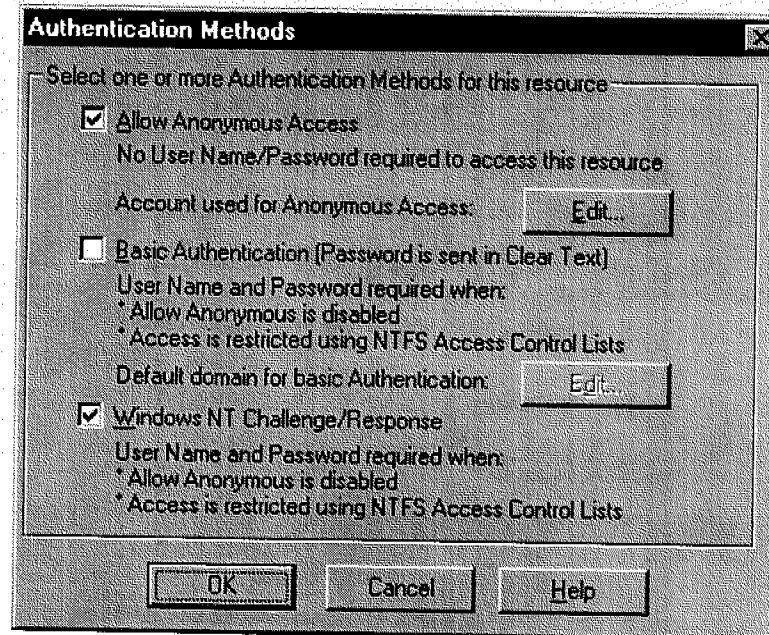


Figure 4.20 Configuring authentication schemes in the Authentication Methods dialog box.

If both the Basic Authentication and Windows NT Challenge/Response options are checked in the Authentication Methods dialog box, it is up to the client's browser to decide which type to use. When the Allow Anonymous Access option is checked, requests are executed in the `IUSR_machine_name` context.

This account can also be changed to some other username if required. For instance, you can specify an account in your domain if you want to allow anonymous requests to access resources across the network. To avoid password confusion (the password in the Windows NT security database must match the password specified in the Management Console), you can tell IIS to synchronize password changes between the two storage locations.

When basic authentication is used and certain resources do not allow anonymous access, the client's browser prompts the user to enter credentials. Most users will tend not to enter their names in the Windows NT domain form of "domain/user." By using the Edit button next to the Basic Authentication option, you can configure the basic authentication scheme to use a specified domain, even if the user does not enter one. When a domain structure does not exist or when you want to use only accounts on your local machine, the default domain for basic authentication should be your server machine's name.

Secure communications The Secure Communications section of the Directory Security page allows you to manage Secure Certificates. Secure Certificates are the part of IIS Setup that enables data encryption. (Secure communications are discussed in Chapter 13.).

IP address and domain name restrictions Figure 4.21 shows the IP Address and Domain Name Restrictions dialog box that appears when you click the associated Edit button on the Directory Security page.

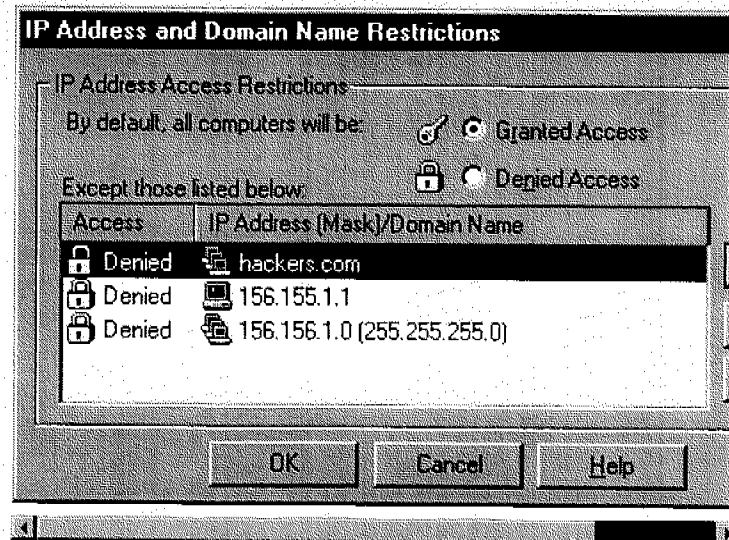


Figure 4.21 Setting restrictions in the IP Address And Domain Name Restrictions dialog box.

In this dialog box, you can set up powerful protection against hackers and other unfriendly users. You can deny (or allow) connections from specific IP addresses or ranges of addresses. In fact, you can even filter connections based on a DNS domain name. The dialog box in the figure shows that nobody in the *hackers.com* domain will be allowed to connect to this Web site. Access is also denied to connections from host 156.155.1.1 and from any host located on the 156.156.1 class B subnet. (Recall the subnet discussion in Chapter 3.)

Note If domain-based filtering is turned on, the server must perform a potentially lengthy reverse DNS lookup for each request. On a busy server, this can slow down performance considerably.

The HTTP Headers property page in the Default Web Site Properties window shown in Figure 4.22 allows you to set content expiration, custom headers, content rating, and MIME mapping for HTTP headers. Chapter 21 discusses HTTP headers in detail, but for now you can consider them a way to send auxiliary information with your responses. For example, the Expires header allows you to indicate how long the data in your response will be valid. You can configure IIS to send such a header by enabling content expiration, as shown in Figure 4.22.

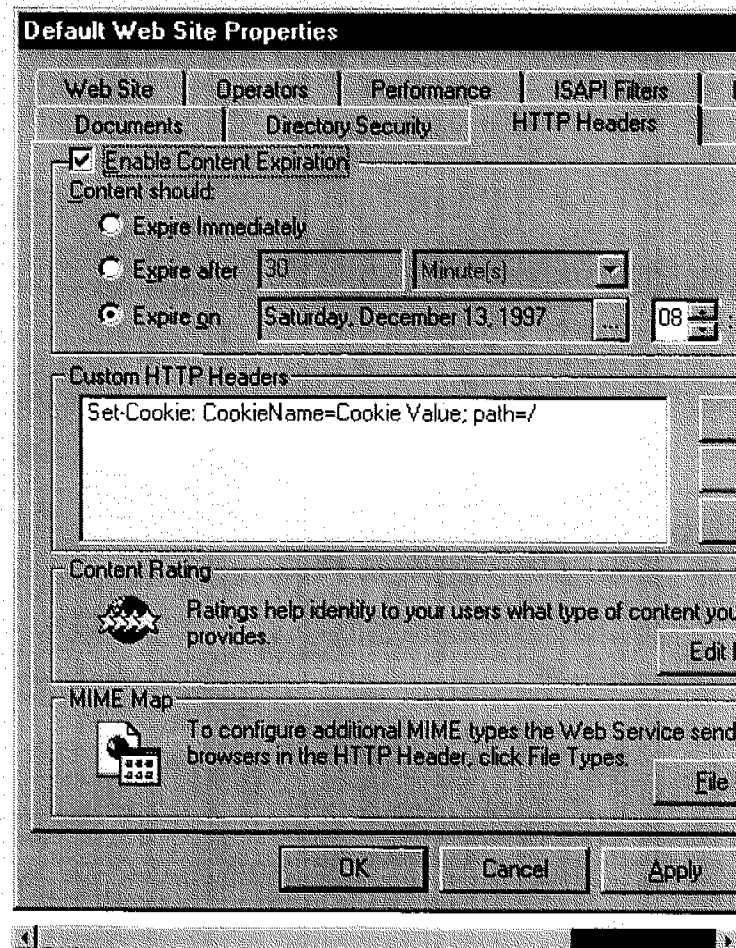


Figure 4.22 The HTTP Headers property page in the Default Web Site Properties window.

The header that results from the configuration in the figure is shown here:

Expires: Sat, 13 Dec 1997 15:32:09 GMT

Notice how IIS is smart enough to convert local Pacific time 8:32 (in the 24-hour system) to Greenwich mean time (GMT). It is also possible to add custom HTTP headers, although not all browsers react to custom headers in a manner that's visible to the user. In the figure, an HTTP Cookie header was added in the Custom HTTP Headers section. To verify proper reception of our custom header, we configured our browser to warn us upon receiving cookies. Figure 4.23 shows that our browser did in fact receive our cookie just as it was configured.

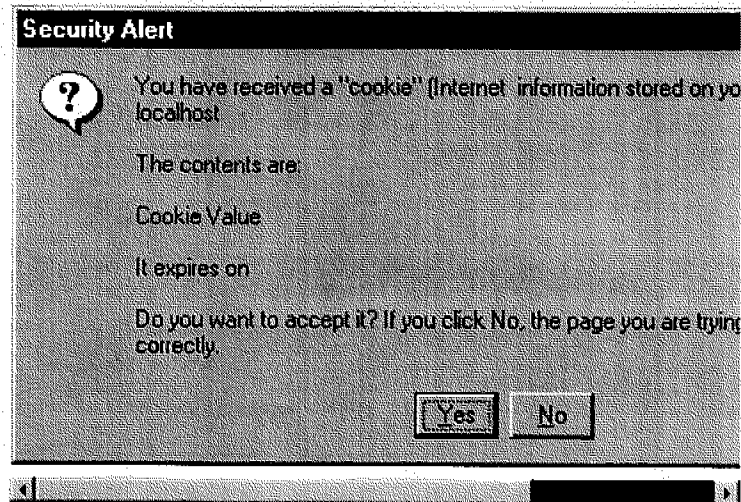


Figure 4.23 The browser receiving a cookie and throwing a dialog box with the warning.

The third section of the HTTP Headers page is Content Rating, a feature supported by some newer browsers. The content rating idea is similar to a motion picture rating that indicates whether the content is appropriate for certain groups of viewers. The Ratings page of the Content Ratings dialog box shown in Figure 4.24 appears when you click Edit Ratings.

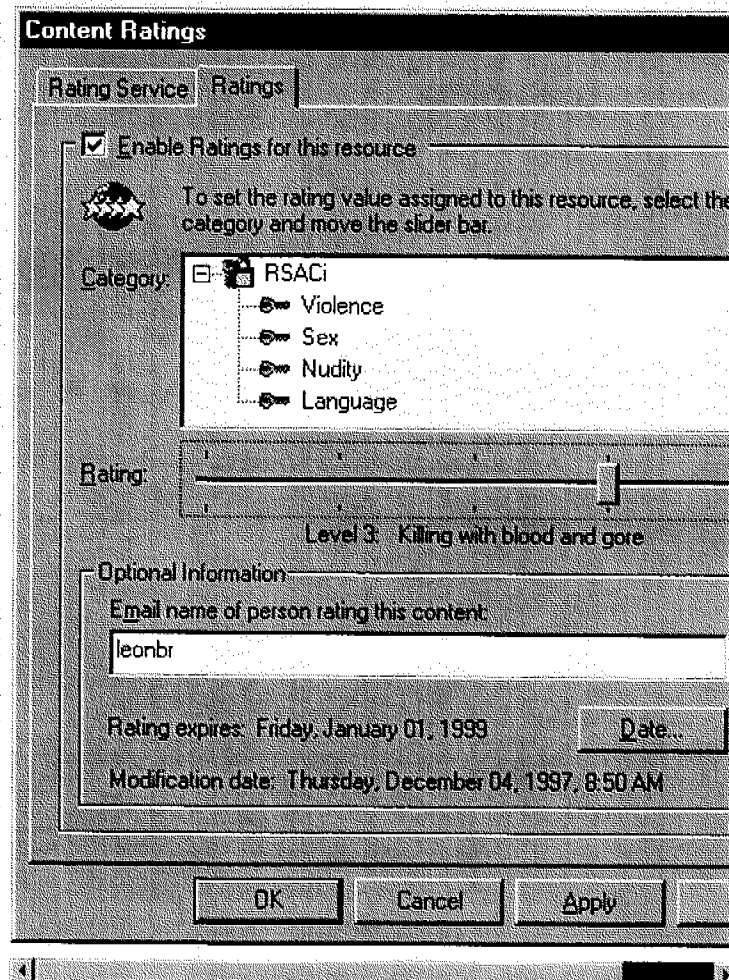


Figure 4.24 The Content Ratings configuration dialog box.

Using the configuration shown in the figure, the server sends a specific header to the browser:

```
PICS-Label: (PICS-1.0 "http://www.rsac.org/ratingsv01.html" 1 by  
"leonbr" on  
"1997.12.04T08:39-0800" exp "1999.01.01T12:00-0800" r (v 3 s 1 n 1  
1 1))
```

Notice again that IIS uses GMT. The browser then interprets the above header to determine whether the content of the site is suitable for viewing by the current user. Pages are rated by their content. The PICS-Label HTTP header above states that this content was rated by leonbr on December 4, 1997. It will expire on January 1, 1999. The available ratings are as follows:

- Violence: Level 3
- Sex: Level 1
- Nudity: Level 1
- Language: Level 1

The browser can be configured to deny access to pages in which ratings exceed preconfigured limits. PICS ratings help guard against kids surfing inappropriate sites. The Rating Service page in the Content Ratings dialog box has links to resources on how to rate Web content. It also allows you to register with the Recreational Software Advisory Council.

Whenever a server sends data to a client, it must indicate what type of data it is sending. Information about the content type helps the browser determine how to handle it. (For instance, it might have to start a helper application.) Information about the type of data being sent from the server to the client is located in an HTTP Content-Type header. The server has a database (which is actually stored in the metabase) of associations of file extensions and their corresponding MIME types. When a request for a specific file is processed, the server finds the correct type based on the file's extension and sends the response with the appropriate Content Type header. Clicking the File Types button in the MIME Map section at the bottom of the HTTP Headers page in the Default Web Sites Properties window (Figure 4.22) opens the dialog box shown in Figure 4.25. Here you can enter custom data types for any file extension.

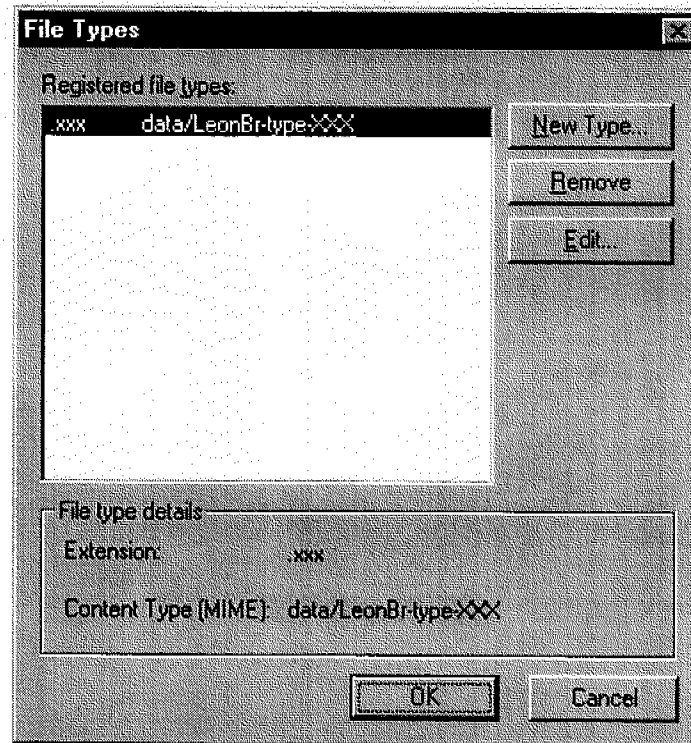


Figure 4.25 The File Types dialog box.

The Custom Errors Page

Last but not least is the Custom Errors property page, shown in Figure 4.26.

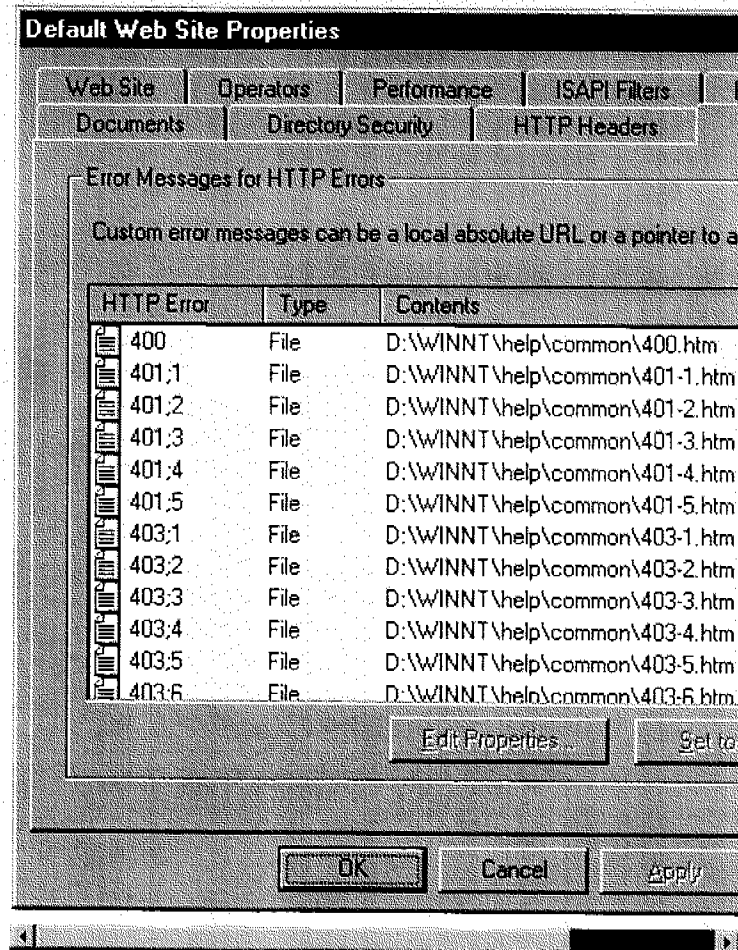


Figure 4.26 Configuring URLs for custom error messages in the Custom Errors page.

IIS's flexibility in reporting errors is one of its best features. Normally, IIS replies to a client's request with the HTTP status code of "200 OK" and data such as HTML text. When an error occurs, for example if a resource could not be found, the server sends an appropriate HTTP status code and possibly a brief error message to the browser. The browser displays the text of the message to help the user determine what went wrong.

From this page, you can configure the location from which the message text associated with the error should come. There are three potential sources:

- The default error message, which can be very short, such as "HTTP/1.1 404 Object Not Found."
- A file with the error message as HTML text. You can create a file with explanations of what went wrong and what needs to be done to fix it. (For instance, for "Object Not Found" errors, you might tell the user to correct the URL and try again.)
- A URL that will provide error information. This can be an HTML file, an ISAPI extension, or even an ASP script that will generate extended error information to aid in troubleshooting. Note that the URL must be on the local machine. If the URL is an executable script, such as an ISAPI extension, you need to make sure that when the DLL

terminates, it preserves the HTTP status code that it was called for (and does not change it to "200 OK").

IIS comes with a set of preconfigured HTML pages for many errors, as shown in Figure 4.26. If you configure your error messages to be generated somewhere else but decide later that you want to go back to the IIS default files, you can simply click the Set To Default button.

This chapter has covered all the configuration options available via the administration user interface. Now that you understand how all the components work together, you should be able to configure almost any option on your server that you want. All the configuration information for IIS is stored in the metabase. Changing metabase values affects the behavior of IIS, just as if you were manipulating the Management Console. Chapter 5 covers the metabase, its layout, and how you can programmatically access it for managing IIS.

About the Authors

Leonid Braginski joined Microsoft in 1995 and presently works on the Internet Information Server team in Microsoft's Developer Support Group. He has coauthored several articles in various computer magazines and is a frequent contributor to Microsoft's Knowledge Base.

Matthew Powell is a support engineer for Microsoft's Developer Support organization in Bellevue, Washington, where he has been helping developers of Microsoft Windows applications with their network and Internet programming questions for the last four years.

The above article is courtesy of Microsoft Press
<http://www.microsoft.com/mspress/>. Copyright 1999, Microsoft Corporation.

We at Microsoft Corporation hope that the information in this work is valuable to you. Your use of the information contained in this work, however, is at your sole risk. All information in this work is provided "as-is", without any warranty, whether express or implied, of its accuracy, completeness, fitness for a particular purpose, title or non-infringement, and none of the third-party products or information mentioned in the work are authored, recommended, supported or guaranteed by Microsoft Corporation. Microsoft Corporation shall not be liable for any damages you may sustain by using this information, whether direct, indirect, special, incidental or consequential, even if it has been advised of the possibility of such damages. All prices for products mentioned in this document are subject to change without notice.

International rights

Last updated November 13, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



Insights and Answers for IT Professionals

[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#)

Chapter 5 - Securing Your Site Against Intruders

Connecting computers to the Internet provides for some very powerful and useful scenarios. Within hours it becomes possible to communicate with millions of people and computers worldwide using Transfer Control Protocol/Internet Protocol (TCP/IP). This broad flexibility imposes a degree of risk—not only can you communicate with people and other systems, it is also possible for users to attempt to initiate communication with your system. Although connecting to servers on the Internet is generally done with good intentions, there are malicious individuals who attempt to infiltrate internal networks. The Windows NT operating system was designed with security in mind.

Your security configuration is crucial for safe operation of your server on the Internet. Although it is unlikely that your site will be maliciously tampered with, Internet servers are available to the general public and there may be some degree of public intrusion. This chapter will help you effectively use Windows NT security and Internet Information Server security at your site. You should understand all of the information in this chapter before connecting your computer to a public network. If you do not understand the information, you should consult Windows NT documentation, an authorized Microsoft Solution Provider, or other source before installing your site on the Internet.

This chapter explains:

- General Windows NT security and how to apply it to your site.
- How Internet Information Server security works.
- How to securely configure the WWW service.

Securely Configuring Windows NT Server

Windows NT provides user-account security and Windows NT File System (NTFS) filesystem security. You can use the topics below as a checklist to ensure you have effectively used User Accounts and NTFS to secure Windows NT Server. Additionally, you can prevent security breaches by properly configuring the services running on your computer.

Preventing Intrusion by Setting Up User Accounts

Windows NT security helps you protect your computer and its resources by requiring assigned user accounts. You can control access to all computer resources by limiting the user rights of these accounts.

Every operation on a computer running Windows NT identifies who is doing the operation. For example, the username and password that you use to log on to Windows NT identifies who you are and defines what you are authorized to do on that computer.

What a user is authorized to do on a computer is configured in User Manager by setting User Rights in the Policies menu. User rights authorize a user to perform certain actions on the system, including the right to "Log on locally," which is required for users to use Internet Information Server services.

Allowing Anonymous Access with the IUSR_computername Account

The IUSR_computername account is created during Microsoft Internet Information Server setup. For example, if the computer name is marketing1, then the anonymous access account name is IUSR_marketing1.

By default, all Microsoft Internet Information Server client requests use this account. In other words, information server clients are logged on to the computer using the IUSR_computername account. The

IUSR_computername account is permitted only to log on locally. No network rights are granted that could allow an unauthorized user to damage your server or its files.

Note The IUSR_computername account is also added to the group Guests. If you have changed the settings for the Guests group, those changes also apply to the IUSR_computername account. You should review the settings for the Guests group to ensure that they are appropriate for the IUSR_computername account.

If you allow remote access only by the IUSR_computername account, remote users do not provide a username and password, and have only the permissions assigned to that account. This prevents hackers from attempting to gain access to sensitive information with fraudulent or illegally obtained passwords. For some situations this can provide the best security.

Requiring a Username and Password

Conversely, if you require "authenticated" clients, users must supply a valid Windows NT username and password.

Basic authentication does not encrypt your username and password before transmission. Basic authentication is encoded only by using UUencode, and can be decoded easily by anyone with access to your network, or to a segment of the Internet that transfers your packets.

Warning Using basic authentication means you that will send your Windows NT username and password unencrypted over public networks. Intruders could easily learn usernames and passwords.

The WWW service also supports the Windows NT Challenge/Response encrypted password transmission. Microsoft recommends only the Windows NT Challenge/Response method of password authentication.

Windows NT authentication, currently supported only by Microsoft Internet Explorer for Windows 95, encrypts the username and password, providing secure transmission of usernames and passwords over the Internet.

With both basic authentication and Windows NT authentication, no access is permitted unless a valid username and password is supplied. Password authentication is useful if you want only authorized individuals to use your server or specific portions controlled by NTFS. You can have both IUSR_computername access and authenticated access enabled at the same time.

Choose Difficult Passwords

The easiest way for someone to gain unauthorized access to your system is with a stolen or easily guessed password. Make sure that all passwords used on the system, especially those with administrative rights, have difficult-to-guess passwords. In particular make sure to select a good administrator password (a long, mixed-case, alphanumeric password is best) and set the appropriate account policies. Passwords can be set by using the User Manager utility, or at the system logon prompt.

Maintain Strict Account Policies

The User Manager utility provides a way for the system administrator to specify how quickly account passwords expire (which forces users to regularly change passwords), and other policies such as how many bad logon attempts will be tolerated before locking a user out. Use these policies to manage your accounts, particularly those with administrative access, to prevent exhaustive or random password attacks.

Limit the Membership of the Administrator Group

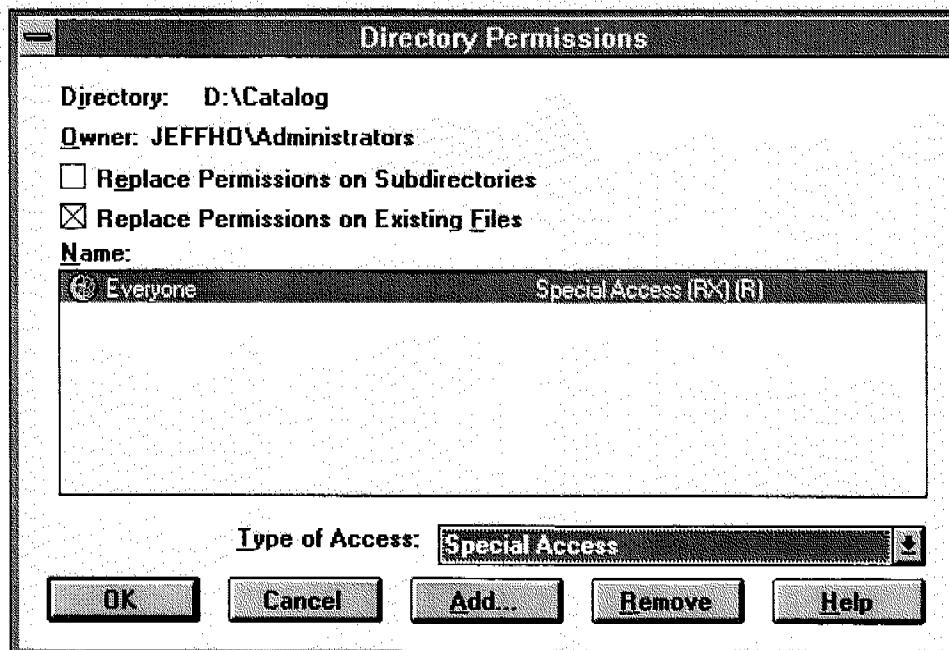
By limiting the members of the Administrator group, you limit the number of users who might choose bad passwords and expose your system.

NTFS File Security

In addition to user accounts, you should place your data files on an NTFS partition. NTFS provides security

and access control for your data files. You can limit access to portions of your file system for specific users and services by using NTFS. In particular, it is a good idea to apply Access Control Lists (ACLs) to your data files for any Internet publishing service.

The NTFS file system gives you very granular control on files by specifying users and groups that are permitted access and what type of access they may have for specific files and directories. For example, some users may have Read-only access, while others may have Read, Change, and Write access. You should ensure that the IUSR_computername or authenticated accounts are granted or denied appropriate access to specific resources.



You should note that the group Everyone contains all users and groups, including the IUSR_computername account and the Guests group. By default the group Everyone has full control of all files created on an NTFS drive.

If there are conflicts between your NTFS settings and Microsoft Internet Information Server settings, the strictest settings will be used.

You should review the security settings for all Microsoft Internet Information Server directories and adjust them appropriately. Generally you should use the settings in the following table:

Directory Type	Suggested Access
content	Read access
programs	Read and Execute access
databases	Read and Write access

Enable Auditing

You can enable auditing of NTFS files and directories on Windows NT Server through the File Manager. You can review the audit records periodically to ensure that no one has gained unauthorized access to sensitive files.

Running Other Network Services

You should review all of the network services that you are using on any computer connected to the Internet.

Run Only the Services that You Need

The fewer services you are running on your system, the less likely a mistake will be made in administration that could be exploited. Use the Services applet in the Windows NT Control Panel to disable any services not absolutely necessary on your Internet server.

Unbind Unnecessary Services from Your Internet Adapter Cards

Use the Bindings feature in the Network applet in the Windows NT Control Panel to unbind any unnecessary services from any network adapter cards connected to the Internet. For example, you might use the Server service to copy new images and documents from computers in your internal network, but you might not want remote users to have direct access to the Server service from the Internet. If you need to use the Server service on your private network, the Server service binding to any network adapter cards connected to the Internet should be disabled. You can use the Windows NT Server service over the Internet; however, you should fully understand the security implications and licensing issues.

The FTP Server service included with Windows NT should also be disabled (this is required if the Microsoft Internet Information Server FTP service will be installed) or configured to ensure adequate security.

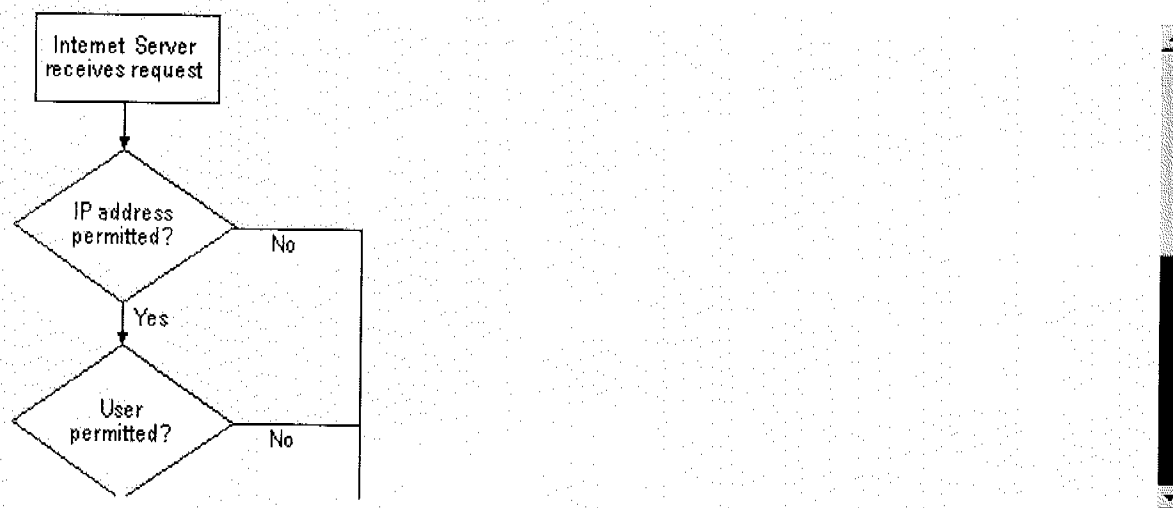
Check Permissions on Network Shares

If you *are* running the Server service on your Internet adapter cards, be sure to doublecheck the permissions set on the shares you have created on the system. It is also wise to doublecheck the permissions set on the files contained in the shares' directories to ensure that you have set them correctly.

How Internet Information Server Security Works

Internet Information Server integrates Windows NT authentication (username and password) security and NTFS file system security. Additional security is implemented by the Internet Information Server by using IP address security and directory access settings.

A simple overview of the security process used on each request is presented in the following illustration.



IP Address Security

The source IP address of every packet received is checked against the Internet Information Server settings in the Advanced property sheet. If Internet Information Server is configured to allow access by all computers except those listed as exceptions to that rule, access is denied to any computer with an IP address included in that list. Conversely, if Internet Information Server is configured to deny all IP addresses, access is denied to all remote users except those whose IP addresses have been specifically granted access.

IP address security is probably most useful on the Internet to exclude everyone except known users. IP address security can also be used to exclude individuals or entire networks that you do not want to grant

access to.

Username Authentication

By default, all requests use anonymous access through the IUSR_computername user account created during Internet Information Server setup. This account is a user account and it granted the right to log on locally.

Username authentication is probably most useful if you want to control access to your server by individual user or group.

Internet Server Permissions

When you assign home and virtual directories for use by Internet Server services, you also specify the type of access that users have for the files in that directory. The WWW service allows you to assign these permissions to a directory:

Read

Allows users to view files contained in a directory.

Execute

Allows users to start applications or scripts. All Internet Server API (ISAPI) applications and Common Gateway (CGI) scripts must be placed into the default \Scripts directory or into a directory configured with Execute permission.

Execute permission must be set in both Internet Service Manager and File Manager when you install applications and scripts on an NTFS drive.

Install server applications into a directory that is configured for Execute in Internet Service Manager and is *not* configured for Write permission on an NTFS drive. This will prevent malicious users from copying programs to your computer that could damage it when run.

Require Secure SSL Channel

Allows users to send information to the server in encrypted format, ensuring data privacy.

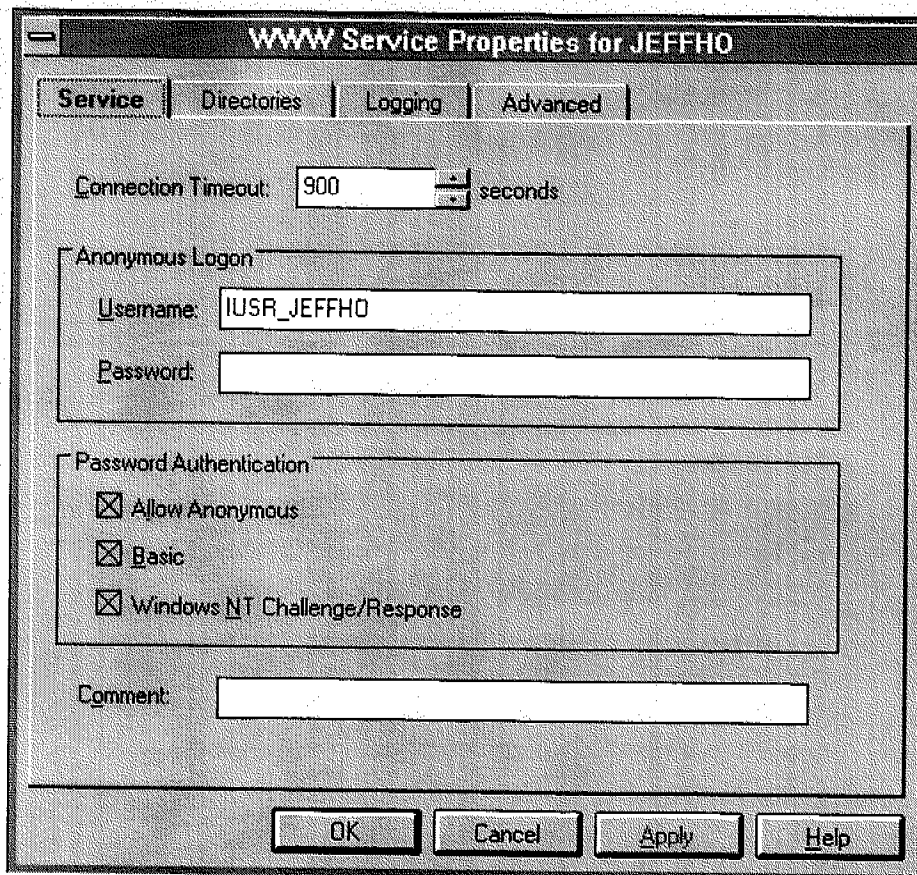
Note that the FTP service supports Read and Write only; the Gopher service supports Read only.

NTFS Permissions

On NTFS drives you must also ensure that similar permissions are set on directories.

Securely Configuring the WWW Service

In addition to implementing the previous suggestions on securing Windows NT Server, you can further enhance your security by using Internet Service Manager to configure the WWW service.



Controlling Access by Username

To gain access to files, users must be identified with a valid username and password.

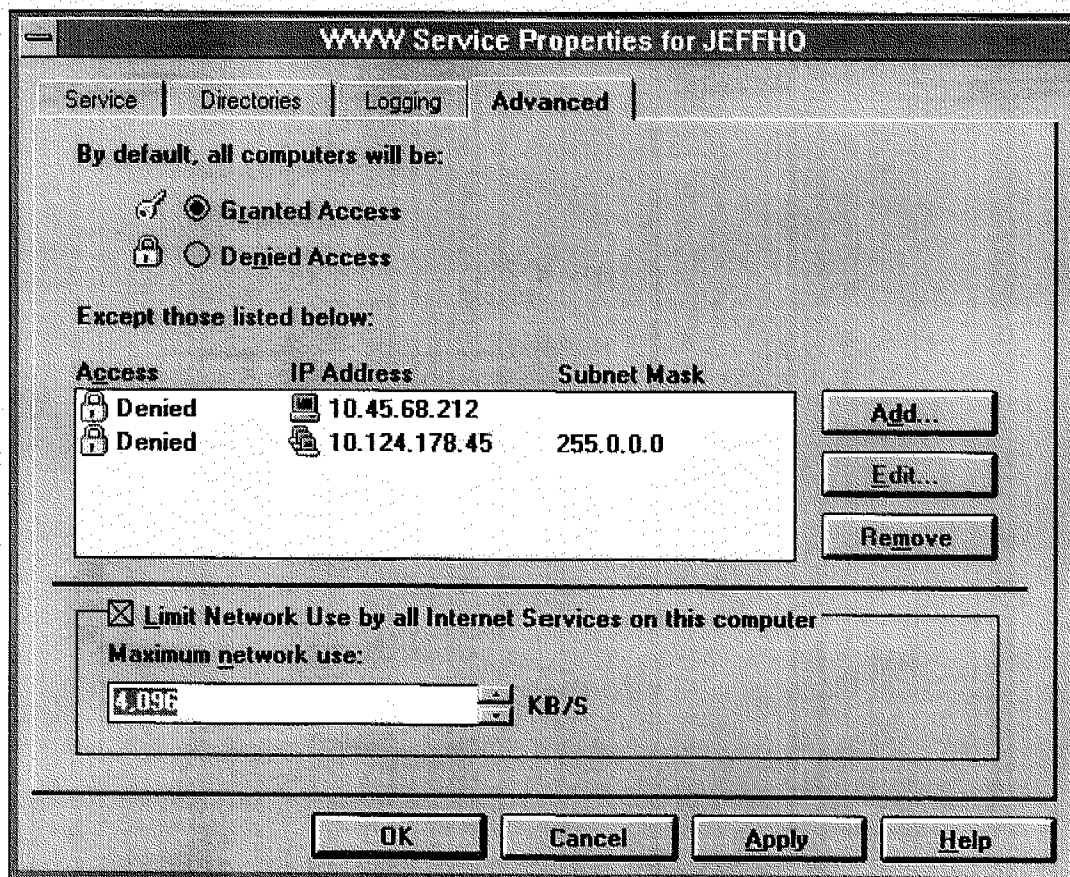
If you are allowing anonymous logon using the `IUSR_computername` account, you should first ensure that computer-wide User Rights (in the User Manager Policies menu) do not allow the `IUSR_computername` account, the Guests group, or the Everyone group any right other than to "Log on Locally." Next, ensure that the file permissions set in the Windows NT File Manager are appropriate for all content directories used by Microsoft Internet Information Server.

If you allow basic or Windows NT Challenge/Response password authentication, users can supply a username and password to gain access to areas that require specific authorization. The usernames must be valid usernames on the computer running Internet Information Server, or in a Windows NT domain accessible from that computer.

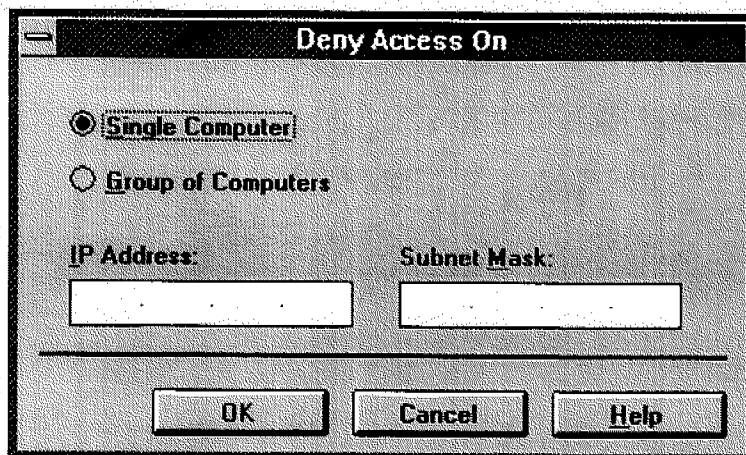
Controlling Access by IP Address

Microsoft Internet Information Server can be configured to grant or deny access to specific IP addresses. For example, you can exclude a harassing individual by denying access to your server from a particular IP address, or prevent entire networks from accessing your server. Conversely, you can choose to enable only specific sites to have access to your service.

Use the Advanced property sheet for the appropriate information service to limit access by IP address.



Choose the Granted Access button or the Denied Access button, and then list the exceptions by using the Add button.



Grant or Deny Access

Choose Single Computer and provide the IP address to exclude a single computer. Choose Group of Computers and provide an IP address and subnet mask to exclude a group of computers.

You are specifying (by IP address) which computer or group of computers will be granted or denied access. If you choose to grant access to all computers by default, you can then specify the computers to be denied access. Conversely, if you choose to deny access to all users by default, you can then specify which computers are allowed access.

Disable Directory Browsing

Unless it is part of your strategy, you should disable directory browsing on the Directories property sheet. Directory browsing exposes the entire file structure; if it is not configured correctly, you run the risk of exposing program files or other files to unauthorized access.

Securing Data Transmissions with Secure Sockets Layer (SSL)

Previous sections of this chapter have dealt with securing your Microsoft Internet Information Server from unauthorized access. This section discusses protocols that use cryptography to secure data transmissions to and from your server.

Microsoft Internet Information Server provides users with a secure communication channel through support for Secure Sockets Layer (SSL) and RSA encryption.

The SSL protocol provides secure data communication through data encryption and decryption. An SSL-enabled server can send and receive private communication across the Internet to SSL-enabled clients (browsers), such as Microsoft Internet Explorer version 2.0 for Windows 95 (included on the Microsoft Internet Information Server compact disc in the \Clients directory).

SSL is a protocol layer between the TCP/IP layer and the application layer (HTTP). SSL provides server authentication, encryption, and data integrity. Authentication assures the client that data is being sent to the correct server and that the server is secure. Encryption assures that the data cannot be read by anyone other than the secure target server. Data integrity assures that the data being transferred has not been altered.

Enabling SSL security on a Microsoft Internet Information Server involves the following steps:

1. Generate a key pair file and a request file.
2. Request a certificate from a Certification Authority.
3. Install the certificate on your server.
4. Activate SSL security on a WWW service directory.

Important Keep in mind the following points when enabling SSL security:

- You can enable SSL security on the root of your Web home directory (\Wwwroot by default) or on one or more virtual directories.
- Once enabled and properly configured, only SSL-enabled clients will be able to communicate with the SSL-enabled WWW directories.
- URLs that point to documents on a SSL-enabled WWW directory must use "https://" instead of "http://" in the URL. Any links using "http://" in the URL will not work on a secure directory.
- SSL security is enabled and disabled by using Internet Service Manager.

How to Acquire an SSL Digital Certificate

The following procedure details the entire SSL configuration process, including how to obtain an SSL digital certificate. You must consult your certificate authority before performing the following steps.

1. Change directories to C:\Inetsrv\Server (or the directory in which you installed Internet Information Server). This is the directory where the key and certificate utilities (Keygen.exe and Setkey.exe) are contained.

2. Use Keygen.exe to create two files. The first file is a key file containing the key pair; the second file is a certificate request file. (Type **keygen** with no arguments to see command syntax and an example).

The following example creates the key file named Keypair.key and the certificate request file named Request.req for a server named www.mycompany.com: The files are generated in the current directory, C:\Inetsrv\Server.

```
C:\inetsrv\server>keygen MyPassword1 keypair.key request.req "C=US, S=Washington, L=Redmond, O=Example, OU=Marketing, CN=www.mycompany.com"
```

```
PCT/SSL Key generation utility, Version 1.0
Copyright (c) 1995 Microsoft Corporation
```

```
Generating key pair of length 1024 bits...
Completed.
```

Send the generated request file, Request.req, to your Certificate Authority for signing.

By default Keygen.exe generates a key pair 1024 bits long. You can use the **-bits** parameter to specify keys that are 512 or 768 bits in length.

The argument in quotation marks in the Keygen.exe command line ("C=US, S=Washington, L=Redmond, O=Example, OU=Marketing, CN=www.mycompany.com") specifies several fields for the certificate request related to your organization and server.

Note Do not use commas in any field. Commas are interpreted as the end of that field and will generate a bad request without warning.

The valid field types follow:

C= 2 Letter ISO Country designations (for example, US, FR, AU, UK, DE)

S= State or Province (for example, Washington, Alberta, or California —do not abbreviate)

L= Locality (for example, Redmond, Calgary, or Redwood City)

O= Organization (Preferably ISO-registered top-level organization or company name)

OU= Organizational Unit

CN= Common Name (Domain Name of server, for example, www.mycompany.com).

If you run Keygen.exe more than once, note that it does not overwrite existing files; instead, it returns an error 80, meaning that the file already exists. Be sure to delete any existing files created by Keygen.exe if you need to run it more than once.

3. E-mail your request to your certificate authority for signing. It is best to include the Keygen.exe command line used, followed by the text from your request file (Request.req). Be sure to remove your password from the command line sent to your certificate authority. For example:

```
From: webmaster@mycompany.com
To: certificates@authority.com
Subject: Certificate Request
```

```
C:\inetsrv\server>keygen <be sure to remove your password> keypair.key request.req "C=US, S=Washington, L=Redmond, O=Example, OU=Marketing, CN=www.mycompany.com"
```

```
-----BEGIN NEW CERTIFICATE REQUEST-----
```

```
MIIBSzcBEQIBADAOMQwwCgYDVQQGEwNVU0EwgZ8WdQYJKoZIhvcNAQEBBQADgY0A
MIGJAoGBaI7n0itueTDEChjJTyOpKPS1DbtRDRouhCeI5Sww2t5fxc7Vs46kPTf9
lJ9UuwpM5TtzqpbBdn7PkpqfV5Cea6LYaAp5U10d8s+IAAqO1Rivvf8az3M8CDUB
eEBbdCWS70a2X9/R44p1oX0DwUnuOnGVw3rh00qgpF0i85bAVVMRagMBAAEWDQYJ
KoZIhvcNAQEEBQADgYEAiCID2qfNkttpx3zagtEEodgDi5VQfA7bSIjXQ0RntKKr
MBA3tsqqNOUDA8KY4Abb7Yr9nFrjf3emSgJ2QcE2NxnEX59NS+JEbLkBTvrt/Twr
3xjU8Wq3sBMuy/9ReozxGwTWQB0RxyhDp3yOncwuSo/N8GUWAB2ddum6+d+LraA=
```

```
-----END NEW CERTIFICATE REQUEST-----
```


4. After completing all documentation requirements from your certificate authority and sending the e-mail in the previous step, you will probably receive an e-mail response containing a signed certificate from your certificate authority. For example:

From: certificate@authority.com
 To: webmaster@mycompany.com
 Subject: Certificate Response

```
-----BEGIN CERTIFICATE-----
ZIICUjCCAb8CBQJyAAL3MA0GCSqGSIb3DQEBAQUAMF8xCzAJBgNVBAYTA1VTMSAw
HsYDVQQKEXdSU0EgrGF0YSBTZWN1cm10eSwgSw5jLjEUMCwGA1UECXM1U2VjdXJ1
IFNlcnZlcjBDZXJ0awZpY2F0aw9uIEF1dGhvcml0eTAeFw05NTA5MTkwMDAwMDBa
FwZanJAzMjAyMzU5NTIamIGFMQSwcQYDVQQGEWJlUzETMBEGA1UECBMKV2FzaGlu
Z3RvbjEQA4GA1UEBxMHUmwkbw9uZDEeMBWGA1UEChMVU1jcm9zb2Z0IENvcnBv
cmF0aw9uMQ8wDQYDVQQLZWZibGRnMjYxHjACBgNVBAMUFUtleu5ldCoubWljcm9z
b2Z0XCvVbTcBnTANBgkqhkiG9w0BAQEFAAOBiWAwgYccGyEAvp3bjApkrNNBtj4q
3ngFdFvMF+Jonem6zwsyBM0WmxvBE0IarmFAK1MAARo9qvqH2LFRdWHHdgb8dhp
h5mzYMTeoriLnY/saoUDu1VMBloUpvh1ErbkNtdVDXoQvWq+Ij5df7y2rQTzF55
uVDNQ8kmcJYDBkAXNSZQbEknPOUCAQMWDQYJKoZIhvcNAQECBQADfgAdT6fQntzx
YXZMsL78qa0heMk+Mb6CKC1ZLBCYQWKSOGZBWfuhpLbokMo8CV3u3/Uck/RxLSzp
XIMU5aDWP6gv8XUraDX1whEAB3fBpDhKQE81nKpcVjiR53UkLGT1jLATYnoCdx9a
HQYCVVSmbSyFKMX4Q5Px00AYd1FOUA==
-----END CERTIFICATE-----
```

5. Copy the text to a file by using Notepad or other text editor and save it (for example, as Certif.txt).
6. Use Setkey.exe to install your signed certificate on the server, for example:

```
setkey MyPassword1 keypair.key certif.txt
```

Note If you do not specify an IP address, the same certificate will be applied to all virtual servers on the system that are configured to use a secure SSL channel for communication. Specify the IP address of a virtual server if the certificate should apply only to that IP address or domain name. For example:

```
setkey MyPassword1 keypair.key certif.txt 10.191.28.45
```

7. Use Internet Service Manager to set the Require Secure SSL Channel option for directories that you want to protect by using SSL. For example:

The screenshot shows the 'Directory Properties' dialog box. The 'Directory' field is set to 'c:\inetrv\secure'. The 'Virtual Directory' tab is selected, with 'Alias' set to 'Logon'. The 'Account Information' section has empty fields for 'User Name' and 'Password'. The 'Virtual Server' section is checked, with 'Virtual Server IP Address' set to '10 .31 .1 .1'. The 'Access' section has 'Read' and 'Execute' unchecked, and 'Require secure SSL channel' checked. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

The configuration in the preceding graphic shows the SSL-protected directory C:\Www\Secure content for the virtual server on IP address 10.191.28.45. To gain access to this content, a client would specify <https://www.mycompany.com/storefront> (note the "https" rather than "http"). Clients must then use the https:// syntax to access any content in the \Storefront directory. Links to content in the \Storefront directory should be changed to use "https" as well. Standard requests (for example, <http://www.mycompany.com/storefront>) will fail.

Suggestions for SSL Configuration and Operation

Microsoft recommends that you use separate content directories for secure and public content (for example, C:\Inetsrv\Wwwroot\Secure-Content and C:\Inetsrv\Wwwroot\Public-Content). It is important to avoid having a server directory not protected by SSL as a parent for a secure directory.

It is suggested that you save your key file (Keypair.key) in a safe place in case you need it in the future. It is a good idea to store Keypair.key on a floppy disk and remove it from the local system after completing all setup steps. Do not forget the password you assigned to the key pair file in step 2.

Setkey.exe Notes

Do not specify a server name when running Setkey.exe on the local computer. After running Setkey.exe, restart the WWW service to enable SSL.

Last updated January 12, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#) |

Managing Access to Resources and Configuring Proxy Server

Topics on this Page

- ▼ [Configuration Principles](#)
- ▼ [Configuring the WinSock Proxy Server](#)
- ▼ [Configuring Proxy Server Security and Authentication](#)
- ▼ [Case Study: Using Proxy as Part of a Proxy Array](#)

M. Shane Stigler and Mark A. Linsenbardt

Chapter 16 from *IIS 4 and Proxy Server 2, 24 seven*, published by Sybex, Inc.

Clearly, one of the two big reasons to use Proxy Server 2 is to control access to resources; that is, to control user access to the Internet and to the actual Internet connection itself. Configuring such access and managing how it works is a relatively simple process, though depending on the level you are configuring access to, it can get rather tedious from time to time. In this chapter, we will consider some of the specifics of Proxy Server 2 configuration, as well as some principles of proxy server configuration that apply to any proxy server product, not just Microsoft's.

Configuration Principles

For the complicated job Proxy Server performs, its setup is very easy and straightforward. The complicated part of Proxy Server is understanding the principle behind what it does. Proxy Server is actually two separate servers working together to perform similar tasks. These servers are separate services under NT but are both controlled through the IIS Service Manager. The first part of Proxy Server is the Web Proxy Server. This is a fully CERN-compatible Web Proxy Server, so any client that adheres to CERN Web Proxy standards can use the Proxy Server Web Proxy to talk to the Internet. Web Proxy clients do not just have to be Windows-based clients. There are no proprietary elements to the Proxy Server Web Proxy. In fact, there is no special software that must be installed on client workstations for them to be able to see the Proxy Server Web Proxy.

For example, UNIX-based systems connected to an NT server running Proxy Server by the TCP/IP protocol can run the UNIX version of Netscape and connect to the Internet through the Proxy Server Web Proxy Server. All flavors of Netscape can be internally configured to see any standard CERN Web Proxy via the TCP/IP protocol.

The second part of Proxy Server is known as the WinSock Proxy Server. It is proprietary to the Windows environment because special WinSock Proxy client software must be installed on workstations in order for them to access the WinSock Proxy Server.

WinSock Proxy works by replacing the local workstation WinSock DLLs with special DLLs that either keep TCP/IP traffic local or remote it over the WinSock Proxy Server for transport to the Internet, depending on the traffic's destination. Only WinSock version 1.1 communications are supported by the WinSock Proxy Server. Microsoft will be updating Proxy Server to support WinSock 2 communications as soon as it can. When support for WinSock 2 standards will be incorporated into Proxy Server is hard to say because the WinSock 2 standards themselves have not been set. It may be that Proxy Server 3 will still only support WinSock 1.1 standards if WinSock 2 standards have not yet been agreed upon.

By adding a new WinSock layer to a system, the WinSock Proxy client software communicates with the WinSock Proxy Server via a special control channel. This avoids TCP/IP port conflicts on the NT Server running Proxy Server. Any Internet server software can be run on the NT machine running Proxy Server because the WinSock Proxy Server itself does not perform broadband TCP/IP port listening as smaller third party WinSock Proxy software, such as WinGate, does. With these thirdparty software packages, the server

side runs by listening to all applicable TCP/IP ports for traffic. The Proxy Server WinSock Proxy Server communicates with clients via a single control channel. The WinSock Proxy clientside software is responsible for listening to specific TCP/IP ports.

By using WinSock Proxy on client workstations, nearly any Internet application (such as email, Newsgroup reader, and FTP client) can operate locally, just as if it were directly connected to the Internet. Because the communication control takes place at the WinSock level, client applications using the WinSock Proxy interface normally need very little special configuration to work correctly.

The SOCKS Proxy works in a manner fundamentally similar to the WinSock Proxy, only inserting itself into communications when necessary because either end of a socket does not support the WinSock interface.

Once you have a grasp of the three faces of Proxy Server, you will know how to best make use of each element on your private LAN. Accessing the configuring controls for these elements is as easy as opening the MMC. While configuring SOCKS Proxy is important, you will only use it occasionally, so we will focus primarily on the configuration of the Web Proxy and the WinSock Proxy.

MMC and the Internet Service Manager

The major prerequisite for installing Proxy Server is to have the IIS server installed first. The Web Proxy part of Proxy Server runs as a sub-service of the WWW service and therefore requires it in order to function. Proxy Server also makes use of the IIS Service Manager to provide an interface for controlling both the Web Proxy and the WinSock Proxy. If you are running IIS 4, as we have suggested, then the IIS Service Manager can be found in the NT Option Pack Microsoft Internet Information Server folder. If you are running IIS 3, the Internet Service Manager can be located in the Administrative Tools (common) folder. You may have more Internet services running on your system. For example, our server includes the NNTP, SMTP, MTS, MSMQ, WWW, and FTP services, as well as the three proxy services.

Note Whichever IIS version you are running, note that Proxy Server installs into its own folder and has its own service manager—in the case of IIS 4, another plug-in to the MMC.

Connecting to Other Servers

NT is a fantastic environment for being able to fully control services running on other NT machines. The IIS Service Manager can be used to control any valid Microsoft Internet service running on another NT machine on the local LAN or even over the Internet. In fact, the Proxy Server installation routine can be used on NT Workstations for just installing the Administration tool, which is just another name for the IIS Service Manager. By installing the Administration tool on an NT Workstation, that workstation can be used to control the Internet services running on NT Servers anywhere on the network. However, the WinSock Proxy cannot be used to remote NetBios traffic, although it is used to control remote NT services. This means that Proxy Server cannot be used to allow LAN workstations to perform network type activities, such as mapping drives or printing to systems on the Internet. Proxy Server does not currently remote NetBios traffic.

The first two buttons on the toolbar are Connect to Server and Search Servers. The Search button will only locate IIS servers running on other NT machines on the private LAN. It won't search the Internet.

The Connect button can be used to connect to other servers, either by machine name or by IP address. A NetBios machine name or the actual IP address to connect to can be entered here. Once connected, the service display area will list all services running on the other machine, as well as services running on the local machine. From that point, all services can be controlled in any way.

Configuring Service Properties

To configure service properties, highlight the service to configure, rightclick, and choose Properties from the resulting pop-up menu. You can also choose Properties from the Action menu.

Authentication Principles

A large part of configuring both the Web Proxy and WinSock Proxy Servers deals with setting up security. This section will give a brief overview of how Proxy Server deals with security. For a full account of Proxy

Server security, please see Chapter 17.

The Web Proxy service uses two levels of security, whereas the WinSock Proxy service uses only one.

The first level of authentication used by the Web Proxy service is login authentication. Since other operating systems can access a Proxy Server Web Proxy, don't rely on the internal Windows login security. CERN authentication is built into the Web Proxy standard. When a client attempts to use the Proxy Server Web Proxy service, it must send a standard HTTP request for access over port 80. Upon receiving this, the Web Proxy Server returns an authentication challenge to the client. Clients that adhere to CERN authentication (such as Netscape and IE 3 or higher) should see a login prompt displayed. Some clients may be configured to send an authentication name and password directly to the Web Proxy without prompting the user. The client must log in with an anonymous login (if that is permitted) or must provide a log in name and password that is present in the NT user database.

If the Web Proxy login is permitted, the authentication name and password used will also be further used by Proxy Server to determine which specific Web Proxy services the user can access (WWW, FTP, and/or Gopher). This is the second level of authentication: protocol-specific access.

The WinSock Proxy service, on the other hand, uses only protocol-specific authentication because the WinSock Proxy client can only run on a Windows platform. It's assumed that the network itself has already taken care of login authentication. Therefore, the WinSock Proxy service can use the internal NT security layer to demand network identification from clients to find out exactly who they are. That information is used to determine protocol-specific access permissions.

Using the WWW Service to Provide Login Access

Because the Web Proxy service runs as a subservice of the WWW service, the login configuration of the WWW server applies to the Web Proxy service. To examine the login setup of the WWW service, follow these steps:

1. Highlight the WWW service in the service list of the IIS Service Manager.
2. Click the Properties button. The WWW Properties dialog box is displayed.
3. If you want to alter which account controls anonymous login permissions, enter a new user account name in the Username field of the Anonymous login section. Even if a password is blank, NT displays a string of 14 asterisks for enhanced security. As you have learned in earlier chapters, when the IIS server is installed, an NT user account is created to handle the rights given to anonymous logins. The name of this account is usually *IUSR_computername* where *computername* is the name of your NT IIS server. This account should not be assigned a password. If this account is given a password, anonymous logins must provide this password for access. The nature of the anonymous login is to use the e-mail address of the requesting user for a password.
4. Note the TCP Port field of the WWW Properties dialog box. This is the field for indicating which TCP port the WWW server listens to. This also affects which port Proxy Server listens to and can be used to great effect when necessary. For example, if you want to run a completely separate WWW server, you can set the IIS Web server to listen to a port other than 80 for network traffic (for example, port 81). Another WWW server can be installed to listen to the traditional port 80. Proxy clients on the LAN can configure their software to talk to port 81, leaving the other Web server to handle available external connections on port 80. There is an option in the Web Proxy configuration for disabling external Web connections to the IIS server, though. When LAN workstations install the WinSock Proxy client software, they will automatically import the correct settings for whatever port you have set the IIS Web server to listen to.

Forms of Access The WWW service supports three forms of login access, all of which you learned about in earlier chapters. Any single one of these forms can apply, or all three can be used simultaneously.

Anonymous When anonymous logins to the WWW service are permitted, anyone can log in without providing any form of authentication. Unlike FTP anonymous access, which requires a login name of *anonymous* and a valid e-mail address given as a password, WWW anonymous access requires no credentials. If the WWW service does not receive any user information upon client connection, it is assumed that the connection should be extended anonymous login access permissions.

Basic (Clear Text) When clear text logins are permitted, WWW clients can present their username and password in a standard, low-level encryption format. This form of authentication is fairly simple to break

and should be avoided if possible.

Windows NT Challenge/Response This option is the highest form of security that the WWW service supports. When a client attempts to access the WWW service, the WWW service demands the presentation of login credentials in NT security encryption and format (NTLM CR). In order for this form of login authentication to be available, the WinSock Proxy service must be present, and the client must have the WinSock Proxy client software loaded. Windows 9X and NT support this form of authentication, but Windows 3.1 and Windows for Workgroups 3.11 do not. However, they can be upgraded to do so if you can still find the software.

The WWW service is designed to field access requests from the outside Internet, and Proxy Server is designed to field access requests from the inside. Keep in mind that a TCP/IP connection to the Internet is just another network connection. Login authentication can take place over a dialup link just as it can over twisted pair cable.

Proxy Server and NT Security

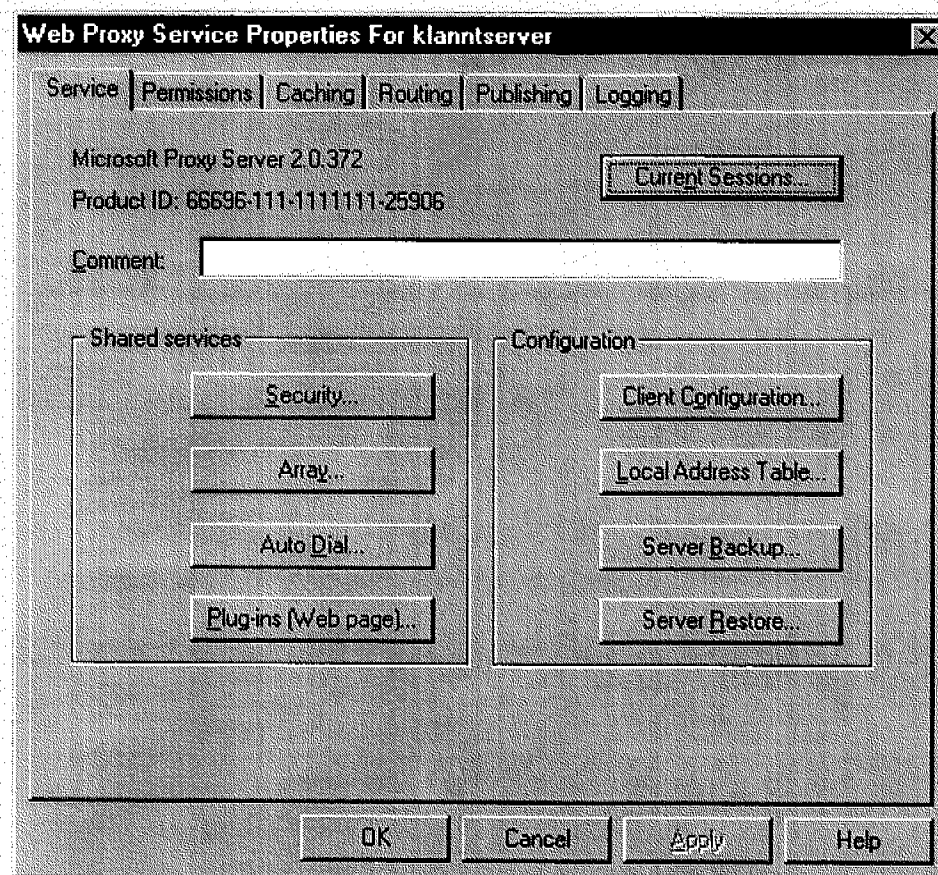
Proxy Server runs as a standard NT service. As such, access by clients is controlled on a user-by-user basis. When a user attempts to connect, Proxy Server consults the internal NT user database. Both the Web Proxy and the WinSock Proxy Server grant access to Internet protocols. A protocol is a TCP/IP virtual port and a standard form of communication between two applications: client and server side. For example, the NNTP protocol is a form of communication between a newsgroup server and a newsgroup reader. By convention, this communication is carried out over TCP port 119. Proxy Server grants outside access on a protocol-by-protocol basis. All communication with the Web Proxy Server happens over port 80, no matter if the client is a WWW client, an FTP client, or a Gopher client. The Web Proxy Server determines the protocol request by the format of the data. The WinSock Proxy Server determines the protocol by the port the client attempted to connect to.

Each protocol handled by the Web Proxy and WinSock Proxy Servers has independent access permissions assigned to them. These permissions can be in the form of permission for specific users or permission for a group of LAN users. Proxy Server can take full advantage of local NT security groups for assigning access to protocols. Out of the box, neither the Web Proxy nor the WinSock Proxy Servers have permissions assigned to any of the protocols they support. Therefore, no one can use Proxy Server until the administrator does some reconfiguring. The following configuration information will cover the basic steps needed. Chapter 17 covers the in-depth issues associated with Proxy Server security.

Configuring the Web Proxy Server

The first thing to do is open up the Web Proxy Server configuration dialog box. To configure the Web Proxy Server, follow these steps:

1. Open the IIS Service Manager.
2. Highlight the Web Proxy Server in the service list.
3. Click the properties button on the toolbar or choose the Properties option from one of the available dialog boxes. Figure 16.1 shows the Web Proxy Server configuration dialog box.



If your browser does not support inline frames, click [here](#) to view on a separate page.

Figure 16.1 The Web Proxy Server configuration dialog box

Conforming to the Microsoft configuration interface format, elements of the Web Proxy configuration are accessed via tabs at the top of the dialog box. The following is a basic description of the purpose of each tab.

Service A basic description of the Web Proxy service can be added on this tab. If only a specific group of users is permitted to access the server, some comment to that effect would be a good idea. A large environment of Internet servers is easier to manage if you know what each server does. This tab also allows you to access the LAT (Local Address Table) and edit it as needed.

Permissions Access permissions for each protocol handled by the Web Proxy Server are configured on this tab.

Caching This tab has settings that control the Proxy Server Web Proxy cache.

Routing This tab has settings that control how the proxy is routed outside the local network, including the aliasing that the proxy performs.

Publishing Computers downstream from the Proxy Server can use it to publish to the Internet. Enabling Web publishing allows you to control how such publishing works.

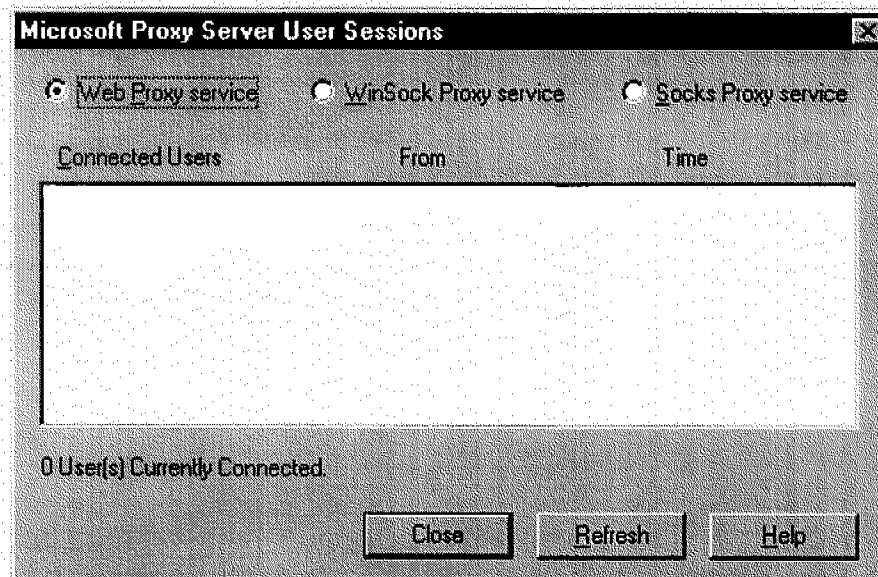
Logging This tab has settings that control how the Proxy Server Web Proxy logs activity information. Tracking access information is second in importance only to security.

To select a tab, click it. The following sections will consider each of the tabs.

The Service Tab

The Web Proxy Server comment will be displayed in the IIS Service Manager service display area. If there are many Internet services running on a network, using comments is important to keep things straight. This tab also allows you to view current connections and edit the LAT.

Viewing Current Sessions At the top, right-hand side of this tab, there is a button for viewing online sessions to the Web Proxy Server. Click this button to view current sessions. Figure 16.2 shows this dialog box.



If your browser does not support inline frames, click [here](#) to view on a separate page.

Figure 16.2 The Microsoft Proxy Server User Sessions dialog box

This dialog box shows the name of the connected user, the IP address that user comes from, and how long that user has been online. The username is *anonymous* if no authentication information has been exchanged between the client and Proxy Server. If anonymous access is not permitted, the username will be displayed. This dialog box does not dynamically refresh itself. To update the list, click the Refresh button. Click the close button to return to the Web Proxy configuration dialog box.

Editing the LAT The LAT is the table that indicates which addresses are local to the network. This is a text file stored in the MSPCLNT share and is transferred to WinSock Proxy clients when the WinSock Proxy client software is installed. This file is named *msplat.txt*. The Edit Local Address Table button calls up an editor that will allow you to make changes to the LAT should your network arrangement change. The LAT is also dynamically sent to clients via the WinSock Proxy control channel.

You can make changes to the LAT in the same manner that you did when Proxy Server was installed. The Construct Table button in the LAT editor will call up another dialog box.

This dialog box allows you to import the values found in the NT routing table to create the LAT. The NT routing table contains all IP information about how to route TCP/IP packets between all network interfaces on the NT Server. There are also options on this dialog box for creating entries in the LAT for the reserved local IP subnets. Chapter 15 covers the details of configuring this dialog box when installing Proxy Server.

Note that the dialog box shows any existing RAS connection of your NT Server as a valid network interface, even though it will be grayed out near the bottom of the dialog box. If you have a static IP for your network connection to an ISP, that static IP address should be part of the LAT.

Once the LAT has been edited correctly, the NT Server should be rebooted in order for the changes to take effect.

The Permissions Tab

Configuring the permissions for the Web Proxy Server protocols is simple compared to configuring permissions for the WinSock Proxy Server. With the Web Proxy Server, only three protocols have to be dealt with. These three protocols also have nothing special to configure. With protocols handled by the WinSock Proxy Server, many more configuration elements are involved, so you have to be a bit more careful in making configuration decisions.

The Enable Access Control check box turns on and off all forms of access restrictions. When not checked, the Web Proxy permits any connections, regardless of the credentials of the client needing access. When checked, the permissions settings restrict client access accordingly.

The drop-down box allows you to select the protocol to configure permissions for. Three Web Proxy protocols can be configured: HTTP, FTP, and Gopher (even though Gopher support is deprecated in IIS 4). The display area shows which NT users or groups have permission to use the indicated protocol. By default, the display area shows no access. If you do not have a need for manual security associated with each protocol, assigning Everyone as a permission to a protocol opens the protocol up for LANwide use.

You can also assign the Everyone group to the Unlimited Access protocol to open the Web Proxy to unlimited access. Your Administrator group should be assigned to the Unlimited Access protocol. This ensures that users with administrator privileges will not be hampered in any way.

Clicking the Add button allows you to select NT users or groups who should have permission to use the protocol.

If the current domain is in a trust relationship with another domain, you have access to add users and groups from the other domain into the permission list for the protocol being configured. The List Names From drop-down list allows you to select the domain from which to draw users and groups. The default is the home domain of the Proxy Server.

To add a user or group to the permission list for the protocol, follow these steps:

1. Highlight the group to add to the permission list. If you want to add a specific user, click the Show Users button. Proxy Server pulls in a list of all users of the selected domain.
2. Click the Add button. The user or group selected shows up in the Add Names display area.
3. Repeat the process and select all users and groups to give permission to for this protocol.
4. Click OK. Those users and groups will now have permission to use this Web Proxy protocol.

The Show Members button allows you to display exactly which NT users are members of a highlighted group. This is very handy to view just who you are granting Web Proxy permission to when you are granting permission to a group.

The Search button allows you to search for a user or a group within the selected domains that can be contacted from the current domain. Multiple domains can be searched simultaneously. On large networks, this is a handy feature. Domains must be in a trust relationship before groups and users can be shared between them.

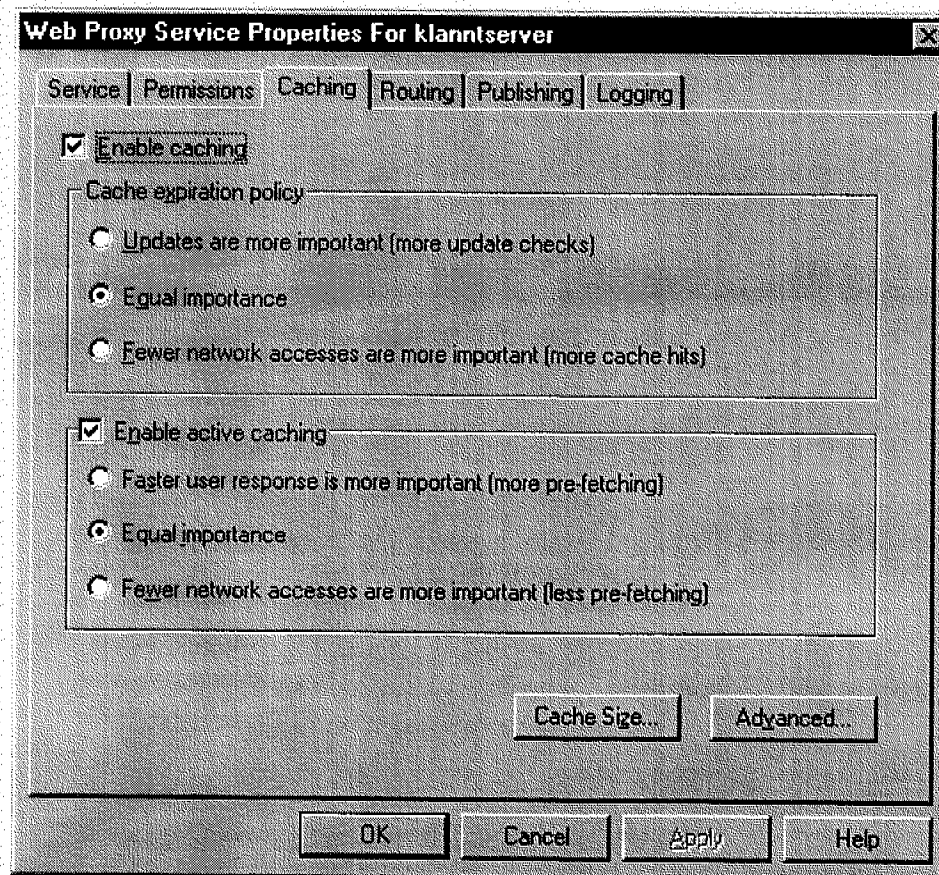
Back on the Permission tab, the Remove button removes a highlighted user or group from the permission list of a protocol.

You may consider creating an Internet group rather than relying on one of the existing NT groups for handling Internet access.

The Caching Tab

The Web Proxy service can cache objects that pass through it on their way to clients. These objects can be graphics, sound files, icons, anything that would normally be part of a Web page. Currently, only WWW objects are cached. Files transferred by the FTP protocol through Proxy Server are not cached, just as Gopher data is not cached. These stored objects can later be issued out to requesting clients on the private LAN if the right conditions are met (such as if the object has not expired or if the object is unchanged on the

server). This reduces the amount of external traffic Proxy Server has to maintain. The Web Proxy cache settings are controlled through this tab, shown in Figure 16.3.



If your browser does not support inline frames, click [here](#) to view on a separate page.

Figure 16.3 The Caching tab of the Web Proxy

Caching can be turned on and off through the Enable Caching check box. Turning the cache off does not mean that Proxy Server will not serve out cached objects to clients. It means that it does not actively store any new incoming objects into the cache.

Modifying the Cache Expiration Policy Objects held within the cache are set to expire after a certain time period. This is called an object's time to live or *TTL*. It is a value measured in seconds. Two things can happen to an object when it has expired:

- The object will no longer be issued out by Proxy Server from the cache to clients, and a new version of the object will be kept when a client requests the object from the Internet.
- Proxy Server will actively update the objects on its own if active caching is configured properly.

To access and control TTL information directly, you must click the Advanced button on the Caching tab. Otherwise, the input you provide to Proxy Server from the radio buttons on the tab only gives it direction in how to make decisions about the appropriate TTL for a given page or site.

Modifying the Active Caching Policy Active caching causes Proxy Server to go out to the Internet and retrieve a fresh copy of an object without needing a client to prompt it to do so. This ensures that popular objects in the cache are always under their TTL and are synched with the originals of the objects on the Internet. This means that clients get HTTP objects locally and do not clutter up the Internet connection.

The Enable active caching check box turns active caching on and off. Proxy Server does not need to be

restarted for any alteration in the active caching policy to take effect.

When you choose the first radio button, Proxy Server caches objects more actively. The active caching implementation is controlled by an advanced algorithm that factors in elements such as object popularity and Proxy Server peak access times. When the algorithm determines it to be the correct time to update an object based on these factors, Proxy Server refreshes the object from the Internet. When you choose the last radio button, the algorithm is adjusted so that active caching occurs less frequently, and the Internet connection is not crowded with Proxy Server caching activity.

Modifying Cache Size and Directories As discussed in previous chapters, Proxy Server's available cache space should be at least 100megs plus 1/2meg for every proxy client that will be supported. If there are many users on a LAN accessing many different sites on the Internet, the suggested size may not be enough to provide adequate caching services. Click the Cache Size button to modify where the cache directories are and how large they should be.

By default, Proxy Server sets up five directories to store cache data. These cache directories are set up under the *URLCACHE* directory. They are named *DIR1*, *DIR2*, *DIR3*, *DIR4*, and *DIR5*. The reason Proxy Server uses multiple cache directories is to speed up access to objects. When a single large cache directory is used, searching the directory for the right object (in the Proxy Server) can be time-consuming. Cache directories should always be placed on a local hard drive, not on a network drive.

Setting the cache is as easy as indicating the drive for a piece, or all, of the cache. The *URLCACHE* directory will be created automatically on all selected drives. Each of the subdirectories within the main *URLCACHE* directory will be used equally by the Web Proxy. The Set button must be clicked to set any size alterations or additions before the changes take effect. Proxy Server does not have to be restarted for any cache change to take effect.

Advanced Cache Settings The Advanced button on the Proxy Service Properties dialog box allows you to control elements, such as what protocols are cached and the maximum size of objects that should be cached, and to filter sites so that their objects are not cached. In Proxy Server's current version, only WWW objects are cached. The ability to enable caching of FTP and Gopher objects is not available.

The settings available on this dialog box limit the size of cached objects, return cache objects when the target site is unreachable, and set filters so that specific sites are not cached. The "Limit the size of cache objects" check box allows you to indicate a maximum size for cache objects. Objects above the size indicated in kilobytes will not be cached by Proxy Server and will always be retrieved from target Web sites. The default is to have all objects cached no matter what their size.

The Return Expired Objects When Site Is Unavailable check box controls whether or not Proxy Server will return objects in the cache if the target site is currently unreachable, but the object's TTL has expired. This allows Proxy Server to simulate a successful connection to a target site even when the objects returned are expired. This can be bad and good for obvious reasons. It's up to you how you want to handle this setting.

The lower portion of this dialog box displays any special filter considerations that you might have configured in Proxy Server. Filters can specifically include or exclude certain sites for or from caching. Sites can be set to Always cached or Never cached. It is possible to set a general never cache policy for a root domain but create a special always cache policy to cache certain sites within that domain. For example, you could set a never cache policy for *.microsoft.com, but create an always cache policy for www.microsoft.com. That way, only objects from www.microsoft.com will ever be cached from the microsoft.com root domain. Cache policies can be set for specific paths on a domain, as well.

You might set www.microsoft.com as a never cache site, but the specific path www.microsoft.com/proxy might be set as an always cache site. Append a site with an asterisk if you want all subpaths from the parent path to be cached. Without the asterisk, only the specific path will be cached.

The Add button will present a dialog box for adding a site filter to the cache configuration. Simply enter a site name in the appropriate format, as indicated in the URL field, and indicate whether the site is to be Always or Never cached. Click OK after configuring a site's caching policy.

The Edit button allows you to edit existing cache policies in the same way that you add a new site policy.

The Routing Tab

You will use the Routing tab with arrays to direct client requests for Internet objects. Requests can be routed through an array to upstream Proxy Server computers or directly to the Internet. You use the Routing dialog box to configure routing for Web Proxy clients.

The Use HTTP Via header appends the name in the text box to the HTTP Via header for proxied requests. Typically, this entry should be the name of the computer on which Proxy Server is installed.

The Upstream Routing options determine whether a client request is sent directly to the Internet or to another Web Proxy Server or array. The default is to use a direct connection to the Internet; however, if you use a Web Proxy, you can also set *abackup route*. In the event that the primary upstream Proxy Server or array is inaccessible, you can specify a backup sequence that the Proxy should use to access the Internet.

The Publishing Tab

Computers downstream from the Proxy Server computer can use Proxy Server to publish to the Internet. Proxy Server supports reverse proxying and reverse hosting. These two features enhance security by allowing any computer on your internal network to publish to the Internet. All incoming and outgoing requests are filtered through the Proxy Server computer. In addition, Proxy Server can also cache incoming requests from the Internet, which provides safe, easy access.

To take advantage of these settings, you must enable Web publishing. Then, you can decide whether incoming Web server requests should be discarded, sent to the local Web server, or transferred to another Web server. If transferred to another Web server, you can specify the request path and the mapping it should use within the spaces at the bottom of the dialog box.

The Logging Tab

Proxy Server keeps a very good record of who uses the Web Proxy or WinSock Proxy services and exactly what sites they access. By default, Proxy Server logs data in a straight text format. Log files are stored in the \WINNT\SYSTEM32\MSPLOGS directory for Web Proxy accesses and in the \WINNT\SYSTEM32\RWSLOGS directory for WinSock Proxy accesses. By default, both services start a new log file daily. The filenames are YYMMDD.LOG where YY is the year, MM is the month, and DD is the day. These log files are prefixed with WS for Winsock and W3 for the Web Proxy.

The following list is a description of each check box on the Logging tab:

Enable Logging Controls whether the Web Proxy service logs information. Unchecked, the Web Proxy service will not keep track of accesses.

Regular Logging Controls whether a full range of information is stored in each log file. If unchecked, only minimal information is stored in the logs. This cuts down on the size of log files.

Verbose Logging Controls whether or not each Internet access is recorded. By default, Proxy Server Web Proxy will only keep information concerning who on the local LAN accesses the Web Proxy Server. Verbose logging will force Proxy Server to record what Internet sites were visited by each user.

Proxy Server can log to a text file or a SQL or ODBC database, provided that these services are present on the network. Checking the Log to File check box tells Proxy Server to log to a standard text file in the appropriate directory. The Daily, Weekly, Monthly, and When File Size Reaches check boxes tell Proxy Server how often to begin a new log file. If you have little Proxy Server activity, a longer logging period is best. The higher the activity, the shorter the turn-around time should be for opening new log files. Be very watchful of your log files. If a client has a great deal of trouble accessing Proxy Server, it will generate an error line for each bad attempt the client makes. Some of the log files on networks we have seen have easily reached and exceeded 250Mb in a single day due to continuous automatic client reconnection attempts. It's a good idea to archive your log files or delete them on a regular basis to conserve disk space.

If you have a need to change the location where Proxy Server stores Web Proxy logs, you can change the contents of the Log file Directory field.

If the Log to SQL/ODBC Database check box is marked, Proxy Server will attempt to connect to a database server to store its log information. This form of logging is slightly slower than writing to a straight text file, but data manipulation for reports and so on is much more powerful. Any installed ODBC (Open Database Connectivity) drivers can be used. Microsoft Access is a common application that installs a full set of ODBC drivers for external applications, such as Proxy Server, to use when attempting to save data in a database format. During installation, Proxy Server can install several types of current ODBC drivers. These drivers will allow Proxy Server to interface with associated database engines for saving log information in the database engine's own format. Proxy Server can log database information to any machine on the network.

The configuration information in this area of the Logging tab is defined as follows:

ODBC Data Source Name (DSN) This field contains the name of the DSN of the database engine to connect to.

Table This is the name of the table within the database that Proxy Server opens to store its log information.

Username This is the username associated with the database table.

Password If the table is password protected, this field contains the correct password to allow Proxy Server to have access to the table.

Once the SQL/ODBC logging fields have been completed, Proxy Server immediately begins logging to the indicated database table. The service does not have to be restarted.

Configuring the WinSock Proxy Server

Configuration of the WinSock Proxy Server is almost an identical process to configuring the Web Proxy Server. The Service, Logging, and Filters tabs are identical in purpose and configuration elements. Refer back to the tab definitions in the Web Proxy Server configuration section earlier in this chapter for details on the settings involved. The following differences in the three tabs apply:

- The Service tab of the WinSock Proxy configuration does not have a View Sessions button. The WinSock Proxy Server cannot view a list of sessions it is currently supporting; instead, you can view the sessions for the WinSock Proxy from the View Sessions dialog box that you saw figure 16.2.
- The Caching tab is not present in the WinSock Proxy configuration. No caching occurs with WinSock Proxy, so this tab does not apply.
- An extra tab is present. This is the Protocols tab and is used to add support for new protocols or edit settings for support on existing protocols.

A major difference in configuration between the WinSock Proxy Server and the Web Proxy Server is in the Permissions tab. Like the Web Proxy Server, each protocol support by the WinSock Proxy Server is assigned a different set of access permissions. Unlike the Web Proxy Server, administrators can define support for new protocols that do not come pre-configured in the WinSock Proxy Server. Remember that nearly any Internet application can communicate with the WinSock Proxy Server. The client software is responsible for listening to local port requests and establishing a link between the client and the WinSock Proxy Server. As long as a port is correctly configured in the WinSock Proxy setup, almost any Internet application can use the WinSock Proxy Server as though it was directly connected to the Internet.

Protocols the WinSock Proxy Server Supports by Default

By default, the WinSock Proxy Server comes pre-configured to handle all major TCP and UDP port communications. Support for TELNET, FTP (non-proxied), NNTP (Network News Transfer Protocol), SMTP (Simple Mail Transfer Protocol), POP3 (Post Office Protocol 3), Finger, RealAudio, VDO Live, and several other common Internet sockets is configured into the WinSock Proxy Server. This means that unless you have special Internet applications that communicate over an uncommon port, you will probably not have to

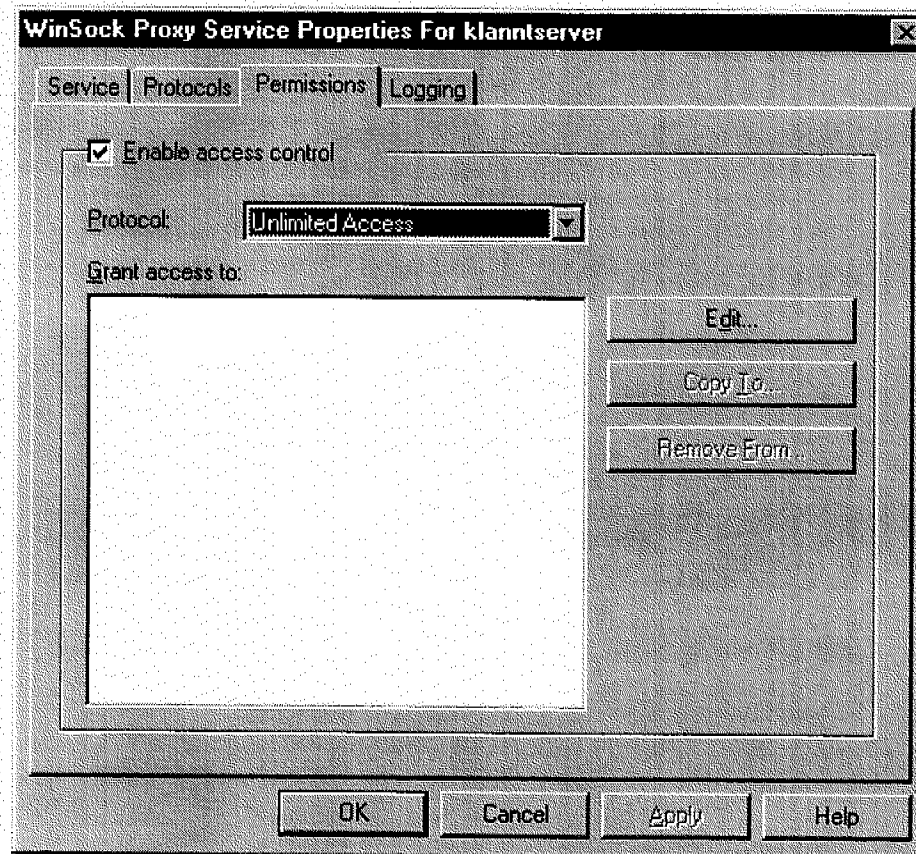
do any special configuration on the WinSock Proxy Server to get all the commonly used client Internet applications running correctly.

The WinSock Proxy Permissions Tab

To open the WinSock Proxy configuration, do the following:

1. Highlight the WinSock Proxy service in the service list of the IIS Service Manager.
2. Click the Properties button on the toolbar.

The WinSock Proxy Permissions tab looks substantially similar to the Web Proxy Permissions tab but provides a greater amount of configurable options than the Web Proxy does. Figure 16.4 shows the WinSock Proxy Permissions tab.



If your browser does not support inline frames, click [here](#) to view on a separate page.

Figure 16.4 The WinSock Proxy Server Permissions tab

Assigning permissions to WinSock Proxy supported protocols is the same process as assigning permissions to Web Proxy protocols. Select the protocol to assign permissions to in the right drop-down list and click the Add button. For more information concerning granting users and network groups rights to use a protocol, refer to the Web Proxy Permission tab discussion earlier in this chapter.

The Copy To and Remove From buttons can be used to copy sets of groups and users from the currently selected protocol to groups of other protocols. For example, if a protocol has seven permission definitions in it, you could display this protocol, select five of these definitions in the traditional Windows select method (holding down CTRL and clicking the desired elements), and then click Copy To. A list of all available protocols would be displayed. You can then select a protocol (or group of protocols with the multiple select method again) and click OK. The selected groups and users are copied into the permission sets of the target protocols. Remove From works in the same way but in reverse. Selected groups and users will be removed

from the selected protocols rather than added to them.

The Protocols Tab

The Protocols tab allows you to modify existing protocols or create new protocols that the WinSock Proxy Server will support. When you view this tab, you see the protocols that the WinSock Proxy Server supports listed in the Protocol Definitions area. From this dialog box, existing protocols can be edited or removed, and new ones can be added. The dialog boxes produced by the Add and Edit buttons are identical.

When adding support for a new protocol, the first thing you have to know in advance is what port the client application talks to its server on and what type of data packets (TCP or UDP) the application uses. Most clients initiate communication with a server over one port, but expect a response over another port. Some clients expect the server to set the return port number, while others expect a return over a consistent port. For example, under normal circumstances, FTP clients initiate communication with an FTP server over port 21 but expect a response over port 23. However, most good FTP clients can be set for something called Passive transfer (PASV Mode), which means that they instruct the server to set up a nonstandard return port.

PASV mode is a security form. It is mostly needed to pass over routers and firewalls. The purpose of a firewall is to prevent access to a network over known ports, such as the return FTP port 23. When the server sets up a non-standard return port, the communication can pass over a firewall.

You must be familiar with how a client/server pair communicates before you can correctly set up the WinSock Proxy Server to connect the two. For example, an application might use UDP packets and initiate communication with conforming servers over port 2417. The application might expect a return channel from the server over a dynamically established port—that is, the return port will vary. The return port is not as important as knowing the initiating port. Most of the time, the WinSock Proxy client software will be able to tell the WinSock Proxy Server what port to expect a return response on from the way the actual client secures the return port on the workstation. The process happens like this:

1. The application initiates communication with a server over port 2417.
2. The WinSock Proxy client software intercepts this call and informs the WinSock Proxy Server about it over its control port—1745.
3. The WinSock Proxy Server begins to communicate with the application as though it was the actual site that the application is trying to talk to. Understand that the WinSock Proxy Server is not responding for the actual target server. It can't. It doesn't know what the application wants. The WinSock Proxy Server only receives the network connection as though it were the target site.
4. The application initiates a listen on a dynamic UDP port for return data from the server it is trying to contact.
5. The WinSock Proxy Server intercepts the listen and tells the WinSock Proxy Server what port the application is listening to.
6. The WinSock Proxy Server initiates a connection between itself and the target site over port 2417. At the same time, the WinSock Proxy Server begins to listen for a response over the UDP port that the application is listening to.
7. When the target site responds on the dynamic return port, the WinSock Proxy Server forwards the response to the application, as if it were the actual site.

In this process, the WinSock Proxy Server acts as the middleman. It pretends to be the target server when talking to the application, and it pretends to be the application when talking to the target server. As long as it knows what port to expect an initial connection on and what type of data packets to toss around, it should be able to handle any Internet client/server combo.

To add support for a new protocol, click the Add button in the Services (protocols) dialog box. The following list defines each element on the Protocol Definition dialog box.

Protocol Name This is any name you want to assign to the protocol.

Initial Connection Port This is the port the client will use when first attempting to contact a server.

Initial Connect Type This can be either TCP or UDP. You must know what type of packets a client uses to initiate communications with a server. If you are not sure, try TCP. TCP packets are more

commonly used than UDP.

Initial Connection Direction This setting tells the WinSock Proxy Server which direction to expect the packets on this port. Since the application begins the communication in the model we have sketched, the direction is outbound. Outbound will be the direction for 95% of all protocols you set up.

Once you have the basics configured, you need to add information about how subsequent connections from the target server back to the client application will be made. With our sample application, we will need to indicate that any UDP port can be used for a return connection. Clicking the Add button (or the Edit button to edit an existing return port) will produce the Return Connection dialog box.

The return port number (or range) should be indicated in the Port or Range fields. A value of 0 indicates that any port may be used as a return port for this protocol.

The Type will set the packet type that is normally the same as the outbound packet type, in this case, UDP. The Direction will be inbound. The application will not send further outbound packets to the target server over a different port. Some protocols may need to send out packets over multiple ports once an initial connection to a server is made. If this is the case, you need to know which ports the client application is utilizing and create multiple subsequent connection entries, or create a range for ports.

Once you have indicated these elements, you can click OK to return to the primary protocol definition dialog box. Those should be the only configuration elements you need to set. For most protocols you will configure, you can set all subsequent connections for any valid port. Once the protocol has been completely defined, click OK to return to the Protocols tab. Don't forget to add permissions to new protocols you configure.

Multiple Proxy Server Gateways

More than one Proxy Server gateway can be used on a network. The Web Proxy and WinSock Proxy Servers behave slightly differently in a network environment where more than one Proxy Server is used.

Multiple Web Proxy Servers

Clients can access multiple Web Proxy Servers in a cascading fashion. Web Proxy Servers can be grouped and accessed in a chain to provide the best possible performance for clients. In order for this to be possible, some form of internal name resolution ability must be present on the network. Either a DNS or WINS server must be available to perform name resolution on behalf of the clients and then provide resolved name information to the clients.

All Web Proxy Servers can be put into an Internet group. This group is defined as an entity by a DNS server or a WINS server. When the group is accessed, either the DNS server or the WINS server serving out the name resolution functionality for the network will sequentially choose a member from the group and resolve the group name requested as the IP of one of the members of the group. The name server is responsible for tracking which member of the group is up for the next resolution request.

Under a WINS environment, a multi-homed, static database entry is created to list all of the Proxy Servers. The WINS server chooses a representative from this list differently from how the DNS server chooses its representative. The WINS server first matches a client's request with the client's IP. The WINS server then tries to find a Proxy Server from the list that has the same subnet as the client. Failing to do that, the WINS server attempts to locate a Proxy Server on the same net as the client. If none of these searches finds a proper candidate, the WINS server picks a member of the group at random and resolves the request to that member's IP address.

Multiple WinSock Proxy Servers

WinSock Proxy Servers cannot be cascaded like Web Proxy Servers can. In order to make best use of multiple WinSock Proxy Servers, network clients should be evenly distributed among all WinSock Proxy Servers to make sure that no one WinSock Proxy Server becomes overloaded. You need to have a good understanding of which Internet protocols demand the most out of a connection. Knowing that will allow you to separate client access correctly. Internet applications, such as RealAudio and VDO Live, consume huge amounts of connection bandwidth and can bring the Internet applications of other network users that are

running through the same connection to a stand still.

Configuring Proxy Server Security and Authentication

Proxy Server security relies directly on the internal security found in NT's architecture. When NT Servers are used in a workgroups-based network, the user information provided on each server is separate and independent. Each server or NT workstation system can maintain a full database of users and groups. These user and group definitions only apply to accessing the particular server on which they are kept.

Arranging a network into a domain takes a little more effort to manage, but the benefits of less confusion and tighter security far outweigh the extra management effort. NT Servers in a workgroup are like islands of independent security. The security credentials needed to access resources or services on one NT Server may not be the same as those needed for a different NT Server.

Login Process

As you should know, several things happen when a workstation logs on to a network. If the workstation is set to logon as a workgroup member, the workstation itself performs user authentication with its own user database of information. If the workstation is set to logon as a domain member, the workstation machine will consult the primary domain controller for user authentication. A login proceeds in the following manner:

1. The domain controller must be found before the logon when the system is started. This process is called *discovery* and is only done when a workstation is set to log on to a domain. The actual method of discovery depends on the protocol(s) the network uses. To discover a PDC (primary domain controller), a workstation must perform a network broadcast, which triggers the PDC of the network to perform its own broadcast to indicate with a directed datagram where the PDC can be found. Once the workstation receives the broadcast response from the PDC, which lets the workstation know exactly where it can find the PDC so that the workstation can correctly direct its own datagrams, the next step of logging on can proceed.
2. Once the PDC is found, the workstation attempts to establish a secure channel between itself and the PDC, or the BDC (backup domain controller) if it responded in place of the PDC. This secure channel consists of datagrams directed back and forth between the workstation and PDC. Each side must prove to the other that they are who they say they are. This process is called Secure Channel Setup.
3. Once the workstation and the PDC have found each other and set up a secure channel, Pass-Through Authentication can occur. In this process, the workstation sends the login username and password to the PDC (or a BDC) in encrypted format. If the user information is correct, the PDC sends back an OK for the workstation to permit the login.
4. After authentication is complete, the system and user are given a security token by the controller that performed the authentication. This token is the actual network item that is passed around to network servers accessed by the client workstation. Any target server will use this token to consult a controller to find out if it is valid and if the associated user should be granted access to use whatever resource the user is attempting to access.

A Proxy Server is like any other resource on the network. Accessing it takes proper network validation. The Proxy Server service is fully capable of utilizing the internal NT security process.

Domain Controllers and Their Impact on Proxy Server

On an NT-based network, the central authority figures are known as *controllers*. There is one primary domain controller and any number of backup domain controllers. These systems are responsible for fielding all Microsoft Network Domain logins and granting or denying access to secured network resources. PDCs and BDCs are always NT Servers and, as such, all share information. User data stored on the PDC is replicated to all BDCs across the domain. The network administrator determines which NT systems are to be BDC machines when these systems are installed. The job of the BDCs is to share some of the workload of the PDC. On medium or large networks, a single authority figure might quickly become overloaded with network traffic. BDCs help to ensure that network performance is kept as high as possible.

Administration of user data can be done from any NT machine, Server, or Workstation, as long as the logged-in user has administrator rights. The main application for modifying user data is User Manager for Domains, which is found in the Administrative Tools folder. When systems are not members of a domain, this application only modifies user information stored in the local user database. When an NT system is a

member of a domain, this application links to an available controllers and modifies the domainwide user database.

When talking about user information concerning Proxy Server authentication, note that we are discussing a domain-wide database of user information. While a Proxy Server machine can be a completely isolated server, not part of any domain, the task of managing separate authorization for network users and Proxy Server users in such an environment becomes far more time-consuming and counterproductive.

In general, most NT installations deal exclusively with a domainbased network. To that end, we will focus on implementing Proxy Server within an NT domain model.

Creating a Global Security Group To create a global group, follow these steps:

1. Click the File menu in the User Manager for Domains.
2. Click the New Global Group selection.
3. The Create Global group dialog box will appear.
4. The name of the group should appear in the Group Name field. In this example, you might name it something like *Proxy Users*.
5. The Description field can be any description you want to give this group.
6. Next, indicate which users should be members of this group. The Not Members list shows all users who are not currently members of this group. Because this is a new group, the Not Members list shows all NT users. Select all users who should be allowed general proxy access and click Add.
7. Click OK. The Proxy Users group is created and a set of users are defined.

Warning Make sure you do not add the *IUSR_servername* user to the group. If this account is added to the group, anonymous users will be granted access to whatever features you assign to the Proxy Users group. This account should only be dealt with on an individual basis and never assigned to any global group.

Once the group is created, it can be used within Proxy Server to define access to various protocols. If this group needs to have special NT network permissions granted to it, the group can be nested within an existing local group that already has the permissions assigned to it. This approach is a simple way of cutting down some of the management time spent on security. If a group of users needs to have certain access permissions in more than one domain, two groups should be created: one that is local and one that is global. Both can have the same name. Users can be assigned to the global group, and the global group can be nested within the local group. The local group can then be granted whatever permissions are needed, and those permissions will filter down to the global group users.

The next step is to grant this group access to a supported protocol, either in the Web Proxy or the WinSock Proxy.

Granting Proxy Permission to the New Group

Open the IIS Service Manager and open the properties for the Web Proxy. Once you have opened the properties of the Web Proxy, select the Permissions tab. By default, no permissions are configured for any protocol in Proxy Server. Therefore, no users have access to get to the Internet through the Web Proxy or the WinSock Proxy. In order to grant access permission to the new Proxy Users group, do the following:

1. Select the protocol you wish to grant access to in the Protocol dropdown list. For example, you will often use the WWW (HTTP) protocol.
2. Click the Add button. This opens a dialog box for adding groups or users to the access list for this protocol.
3. The List Names From drop-down list allows you to select any domain you currently have access to. Access to foreign domains can be through a trust relationship or from having a parallel account in other domains. By default, you can select users and groups from the local domain.
4. The Default Only drop-down list has both local and global groups. However, you can list users by clicking the Show Users button. This displays the users of the domain, as well as the groups. Configuring individual users is fine for small networks or special cases, but this can be a management nightmare for medium or large networks. You should always work with groups whenever possible.
5. Scroll down the name list until the Proxy Users group is displayed.
6. Highlight the Proxy Users group and click the Add button. This adds the Proxy Users group to the Add Names list.

7. You can select any additional groups or users to grant WWW access permission to if necessary.
8. Click OK to return to the Permissions tab. The group appears in the Grant Access To list area and has access to use the WWW protocol.

The Members button on the Add Users and Groups dialog box displays a list of users for the currently highlighted group. If more than one group is selected, this button is not available. The Add button at the bottom of this dialog box will add the group to the Add Name list on the Add Users and Groups dialog box. It is not for adding additional users to the group. This function allows you to view which users are members of the group.

The Search button on the Add Users and Groups dialog box will let you search for users or groups on the local domain or on domains that you have access to, either through a trust relationship or by having a parallel account on the other domain(s).

In this dialog box, you can indicate which domains to search and the name of the user or group you want to search for. You can search in the local domain or in all available domains. By default, all domains will be searched. Search results will be displayed in the lower area. Elements of the search result can be selected. Click the Add button to add the user or group to the permissions list.

Once you have added the Proxy Users group to the permission list for the WWW protocol, the users of that group will be able to use Web browsers through Proxy Server Web Proxy to access WWW sites on the Internet.

Complete this process for all of the protocols (WWW, FTP, Gopher, or Secure) you need to grant users permissions to. The process for adding permissions to WinSock Protocols is very similar, but the WinSock Proxy has special universal access settings that make it easier to grant global protocol permissions for a group of users.

Controlling Inbound Access from the Internet

When Proxy Server is installed, two elements of NT are altered so that security is enhanced. The first element that is altered is IP Forwarding. IP Forwarding is found within the TCP/IP settings. It is turned off by default. It controls whether or not NT will forward IP packets between network interfaces in managers (such as a network card and a RAS connection to an Internet Provider). Under conditions where a dedicated, full-time Internet connection is available to a network and each workstation on the LAN is configured for its own direct Internet access, IP forwarding must be enabled for workstations to pass their packets to the Internet and vice versa. This in itself will halt all inbound traffic at the NT Server, which is connected to the Internet.

To further restrict access to the NT Server from clients connecting from the Internet, Proxy Server disables listening on all TCP/IP ports which do not have permissions set for them. This means that any Internet server application (such as an FTP server, a telnet server, or a POP3 server) running on the connected NT Server will be unable to hear any external inbound traffic until permissions are set for the associated protocol in the WinSock Proxy. The Web Proxy only listens to port 80 for traffic. If permissions are set for any of the supported protocols in the Web Proxy, port 80 will be listened to for inbound traffic.

Isolating Proxy Server on Its Own Domain

If you want to set your network security at a very high level for proxy access, one approach is to set up the NT Server running Proxy Server as a primary domain controller of its own domain. A oneway trust relationship can be established between the Proxy domain and the network domain. The Proxy domain would be set to trust the network domain, but the network domain would not trust the Proxy domain. This arrangement will further limit the access that can take place between the proxy server and all other systems on the network domain.

This arrangement also works well when the network is not set as a domain but rather as a workgroup. The NT Server running Proxy Server can be set on a primary domain controller of its own domain, which will give greater security control and allow easier expansion for future growth.

Case Study: Using Proxy as Part of a Proxy Array

Often, companies find that they have to divide their network into several local groups, using routers and bridges to divide it up. This reduces the amount of traffic that transits certain parts of the physical network.

In such an environment, building a Proxy Server array can be a useful technique, particularly if the users in the different subnets tend to access substantially different pages. In such a case, building a Proxy Server array can speed user access without substantial impact on the network as a whole.

The Problem

Assume for the moment that a company has five divisions, each of which works closely with extranets of vendors and other related organizations. However, none of the different divisions overlap on access to any of these particular extranets. Instead, these only overlap on generic Web site surfing.

The Solution

By configuring a proxy array and setting individual routing tables, you can set up each individual Proxy Server to cache the information appropriate only to its department, while accessing a central Proxy Server for generalized Internet access. Such a construction will speed departmental access to common sites, reduce network traffic, and still maintain our Proxy Server goals of providing central points of management accessible by the administrative staff.

Summing It Up

While Proxy Server lends itself to simple Internet access distributions, using it in a complex environment with a multiple-server array model can help you effectively manage Internet access, as well as network bandwidth.

About the Authors

M. Shane Stigler, MCSE and MCT, is a senior partner in a consulting firm based in Las Vegas, Nevada and provides technical training to a variety of companies.

Mark A. Linsenbardt, MCSE and MCT, is a seasoned trainer who has taught certification classes and provided consulting services all over the country. Currently, Linsenbardt is the president of a small consulting firm in Las Vegas, Nevada.

Copyright © 1999 Sybex, Inc.

We at Microsoft Corporation hope that the information in this work is valuable to you. Your use of the information contained in this work, however, is at your sole risk. All information in this work is provided "as-is", without any warranty, whether express or implied, of its accuracy, completeness, fitness for a particular purpose, title or non-infringement, and none of the third-party products or information mentioned in the work are authored, recommended, supported or guaranteed by Microsoft Corporation. Microsoft Corporation shall not be liable for any damages you may sustain by using this information, whether direct, indirect, special, incidental or consequential, even if it has been advised of the possibility of such damages. All prices for products mentioned in this document are subject to change without notice. International rights = English only.

International rights = English only.

last updated April 24, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use



TechNet Home | Site Map | Events | Downloads | Personalize | Worldwide | Advanced Search |

Chapter 3 - Configuring and Managing Your Internet Information Server

Once you have completed setup and tested your installation, you can use Microsoft Internet Service Manager and other tools to configure the more advanced features of the Internet Information Server services. This chapter tells you how to:

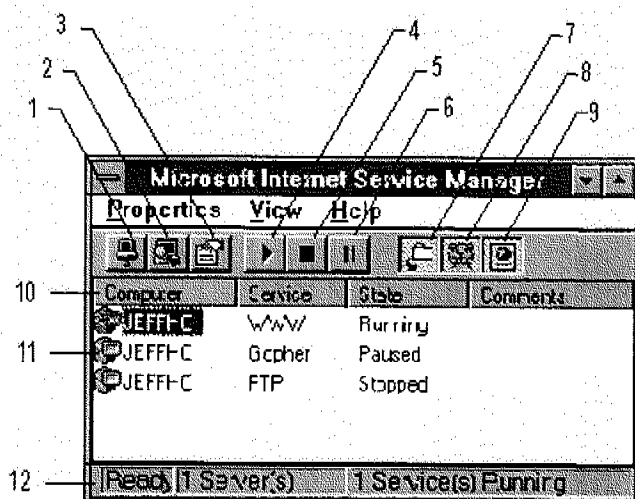
- Configure access permissions for remote clients.
- Establish logon requirements for remote clients.
- Specify home directories and other virtual directories.
- Create multiple virtual servers on a single computer (WWW service only).
- Require content encryption.
- Configure logging options.
- Specify other default settings.
- Use other Windows NT tools to configure or use Internet Information Server.

Microsoft Internet Service Manager

You can use Internet Service Manager to enhance the configuration and performance of your server. Internet Service Manager helps you configure and monitor all the Internet services running on any Windows NT Server-based computer in your network. There are three views available in Internet Service Manager: Reports, Servers, and Services.

Report View

Report view is the default view. Report view alphabetically lists the selected computers, with each installed service shown on a separate line. Click the column headings to alphabetically sort the entire list. Report view is probably most useful for sites with only one or two computers running Internet Information Server. The following illustration lists the functions of the buttons and icons in Internet Service Manager; you can also use the Properties and View drop-down menus for the same functions.



Connect to servers and view property sheets

1. Connects to one specific Internet server.

2. Finds all Internet servers on the network.
3. Displays property sheets to configure the selected service.

Start, stop, or pause a service

4. Starts the selected service.
5. Stops the selected service.
6. Pauses the selected service.

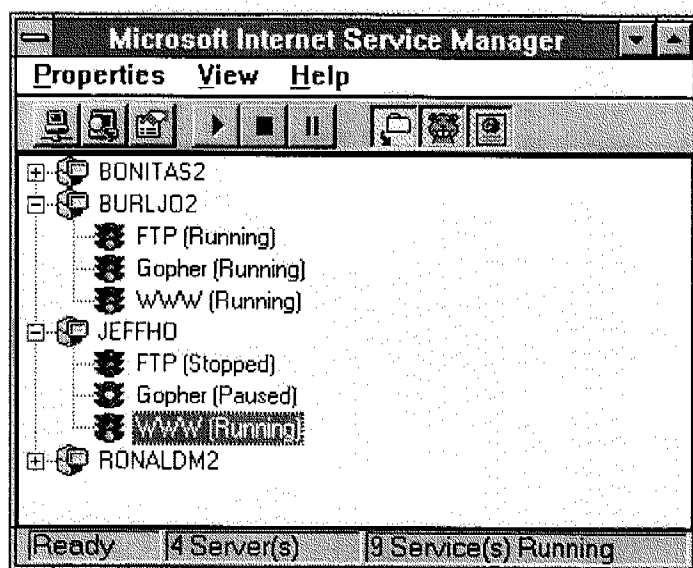
Select which services should be displayed

7. Displays the FTP service in the Internet Service Manager main window.
8. Displays the Gopher service in the Internet Service Manager main window.
9. Displays the WWW service in the Internet Service Manager main window.

Make any necessary adjustments to services

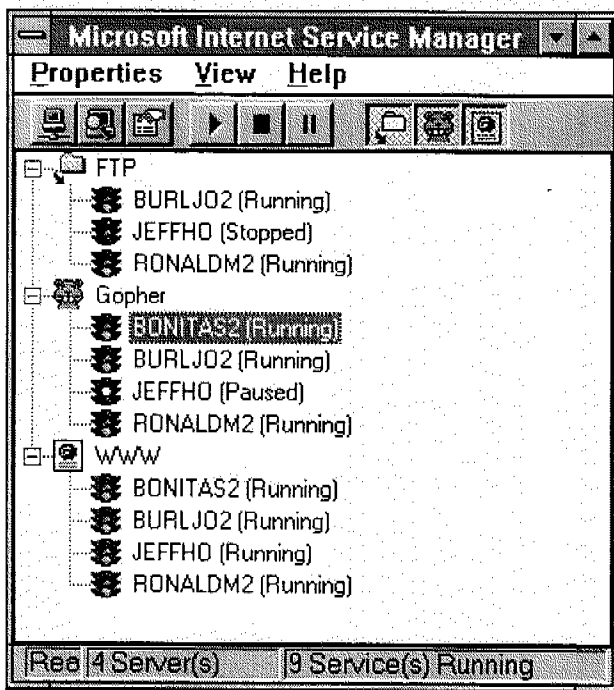
10. Sorts the listings when you click a column heading.
11. Displays the property sheets for a service when you double-click it.
12. Displays server and service status.

Servers View



Servers view displays services running on network servers by computer name. Click the plus symbol next to a server name to see which services that server is running. Doubleclick a service name to see its property sheets. Servers view is most useful for sites running multiple computers when you need to know the status of the services installed on a specific computer.

Services View



Services view lists the services on every selected computer grouped by service name. Click the plus symbol next to a service name to see the servers running that service. Double-click the computer name under a service to see the property sheets for the service running on that computer. Services view is most useful for sites with widely distributed servers when you need to know which computers are running a particular service.

Property Sheets

The Internet Service Manager property sheets can be used to configure and manage the World Wide Web (WWW) and other services. The following information focuses on the WWW service, the most commonly used service.

In Internet Service Manager, double-click a computer or a service name to display its property sheets. Click the tab at the top of each property sheet to display the properties for that category. After setting the properties for the service, click OK to return to the main Internet Service Manager window. Detailed information about each property sheet is included in later chapters on security, directories, and logging.

Note In special instances you may need to use Registry Editor (Regedt32.exe) to configure Internet Information Server or Windows NT Server. See Help for information on Registry entries and when you need to use them.

The Service Property Sheet

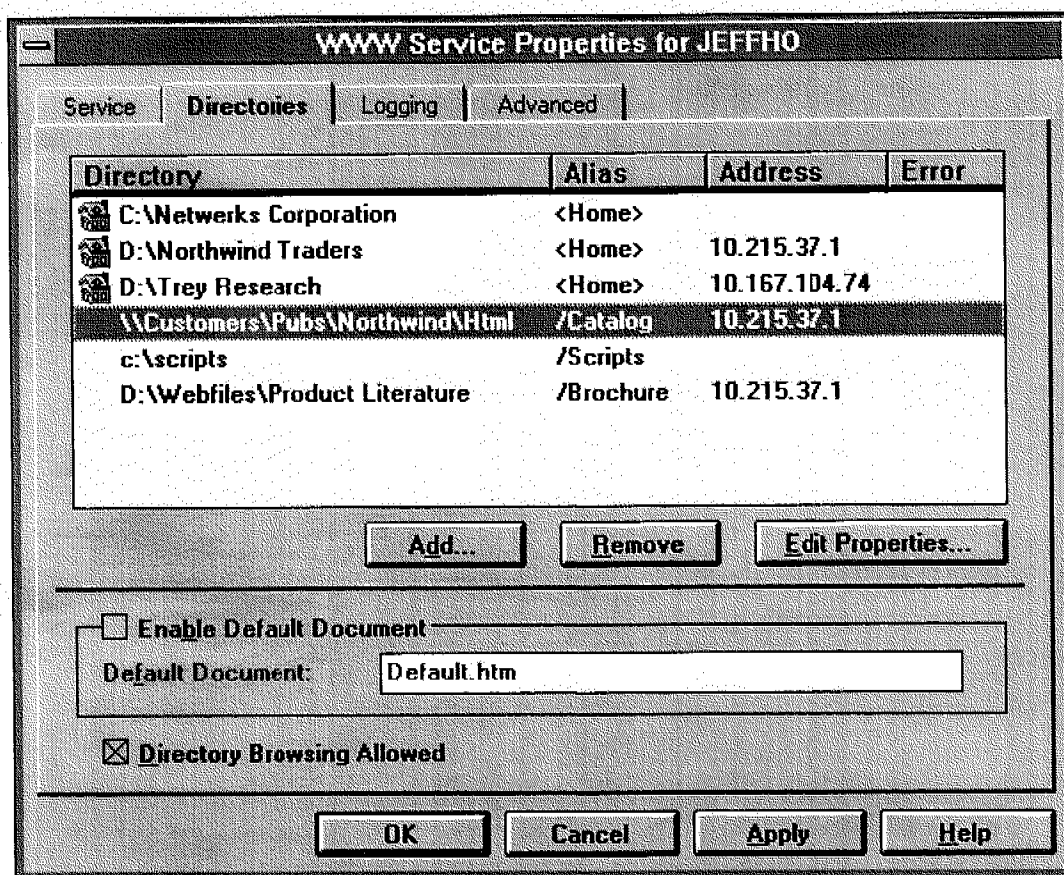
The screenshot shows the 'WWW Service Properties for JEFFHO' dialog box with the 'Service' tab selected. The 'Connection Timeout' is set to 900 seconds. The 'Anonymous Logon' section has 'Username' set to 'IUSR_JEFFHO' and 'Password' is empty. The 'Password Authentication' section has three checked options: 'Allow Anonymous', 'Basic', and 'Windows NT Challenge/Response'. A 'Comment' field is empty. At the bottom are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

WWW Service Properties for JEFFHO	
Service	Directories Logging Advanced
Connection Timeout: 900 seconds	
Anonymous Logon	
Username:	IUSR_JEFFHO
Password:	
Password Authentication	
<input checked="" type="checkbox"/>	Allow Anonymous
<input checked="" type="checkbox"/>	Basic
<input checked="" type="checkbox"/>	Windows NT Challenge/Response
Comment:	
OK	Cancel Apply Help

You use the Service property sheet to control who can use your server and specify the account used for anonymous client requests to log on to the computer. Most Internet sites allow anonymous logons. If you allow anonymous logons, then all user permissions for the user, such as permission to access information, will use the `IUSR_computername` account. To use your current security system to control information access, change the anonymous logon account to existing accounts on your network.

This property sheet also sets the comment in the main Internet Service Manager window. See Chapter 5, "Securing Your Site Against Intruders," for more information.

The Directories Property Sheet



Directory Window

The Directories property sheet lists directories available to users, with some possible exceptions. For more detailed information, see Chapter 6, "Planning Your Content Directories and Virtual Servers."

Default Document and Directory Browsing

The Default Document and Directory Browsing settings in the Directories property sheet for the WWW service are used to set up default displays that will appear if a remote user does not specify a particular file. Directory browsing means that the user is presented with a hypertext listing of the directories and files so that the user can navigate through your directory structure.

You can place a default document in each directory so that when a remote user does not specify a specific file, the default document in that directory is displayed. A hypertext directory listing is sent to the user if directory browsing is enabled and no default document has been provided.

Directory Properties

Directory:

☐ Home Directory

☒ Virtual Directory

Alias:

Account Information

User Name:

Password:

☒ Virtual Server

Virtual Server IP Address:

Access

☒ Read ☐ Execute

☐ Require secure SSL channel

Internet Server provides a default home directory for each service of the primary computer: \Wwwroot, \Gopheroot, and \Ftproot. The files that you place in the home directory of Internet Information Server and its subdirectories are available to remote browsers. You can change the location of the default home directory.

You can also add other directories outside the home directory that will appear to browsers as subdirectories of the home directory. That is, you can publish from other directories and have those directories appear to reside within the home directory. Such directories are called "virtual directories."

The administrator can specify the physical location of the virtual directory and the virtual name, which is the directory name used by remote browsers.

The published directories can be located on local or network drives. If the virtual directory is a network drive, provide username and password with access to that network drive.

Important If you specify a username and password to connect to a network drive, all Internet Information Server access to that directory will use that username and password. You should use care when using Universal Naming Convention (UNC) connections to network drives to prevent possible security breaches.

WWW Virtual Servers

You can have multiple domain names on a single Internet Information Server-based computer so that it will appear that there are additional servers, or "virtual servers." This feature makes it possible to service WWW requests for two domain names (such as <http://www.company1.com/> and <http://www.company2.com/>) from the same computer. Enter the IP (Internet Protocol) address for the home directory, and virtual directories for each virtual server you will create.

If the path for a virtual directory is a network drive, provide a username and password with access to that network drive.

If you have assigned more than one IP address to your server, when you create a directory you must specify which IP address has access to that directory. If no IP address is specified, that directory will be visible to all virtual servers.

Important The default directories created during setup do not specify an IP address. You may need to specify IP addresses for the default directories when you add virtual servers.

See Chapter 6, "Planning Your Content Directories and Virtual Servers," for more information.

The Logging Property Sheet

The screenshot shows the "WWW Service Properties for BOBDA3" dialog box with the "Logging" tab selected. The "Enable Logging" checkbox is checked. Under "Log to File", the "Automatically open new log" checkbox is checked, and the "Monthly" radio button is selected. The "Log file directory" is set to "C:\WINNT35\System32\LogFiles" and the "Log file name" is "INyymm.log". Under "Log to SQL/ODBC Database", the "ODBC Data Source Name (DSN)" field is empty, and the "Table", "User Name", and "Password" fields are also empty. The "OK", "Cancel", "Apply", and "Help" buttons are at the bottom.

Service	Directories	Logging	Advanced
Enable Logging			
<input checked="" type="radio"/> Log to File			
<input checked="" type="checkbox"/> Automatically open new log			
<input type="radio"/> Daily			
<input type="radio"/> Weekly			
<input checked="" type="radio"/> Monthly			
<input type="radio"/> When file size reaches:			
4 MB			
Log file directory: C:\WINNT35\System32\LogFiles			
Log file name: INyymm.log			
<input type="radio"/> Log to SQL/ODBC Database			
ODBC Data Source Name (DSN)			
Table			
User Name			
Password			
OK Cancel Apply Help			

The services can log server activity. Logging provides valuable information about how a server is used. You can send log data to files or to an Open Data Base Connectivity (ODBC)-supported database. If you have

multiple servers or services on a network, you can log all their activity to a single file or database on any network computer.

If you want to log to a file, you can specify how often to create new logs and which directory put the log files in.

If you log to an ODBC data source, you must specify the ODBC Data Source Name (DSN), table, and valid user name and password to the database.

See Chapter 7, "Logging Server Activity," for more information.

The Advanced Property Sheet

The screenshot shows the 'WWW Service Properties for JEFFHO' dialog box with the 'Advanced' tab selected. The 'Service' tab is also visible. The 'Advanced' tab contains the following options:

- By default, all computers will be:**
 - ☒ **Granted Access**
 - ☐ **Denied Access**
- Except those listed below:**
- | Access | IP Address | Subnet Mask |
|--------|---------------|-------------|
| Denied | 10.45.68.212 | |
| Denied | 10.124.178.45 | 255.0.0.0 |
- Buttons:** Add..., Edit..., Remove
- ☒ **Limit Network Use by all Internet Services on this computer**
- Maximum network use:** 4096 KB/S
- Buttons:** OK, Cancel, Apply, Help

You can use Internet Service Manager to prevent access by certain IP addresses to block individuals or groups from gaining access to your server. You can also set the maximum network bandwidth for outbound traffic, to control (throttle) the maximum amount of traffic on your server.

IP Access Control

You can control access to each Microsoft Internet Information Server service by specifying the IP address of the computers to be granted or denied access.

If you choose to grant access to all users by default, you can then specify the computers to be denied access. For example, if you have a form on your WWW server and a particular user on the Internet is entering multiple forms with fictitious information, you can prevent the computer at that IP address from connecting to your site. Conversely, if you choose to deny access to all users by default, you can then specify which computers are allowed access.

Limiting Network Use

You can throttle your Internet services by limiting the network bandwidth allowed for all of the Internet services on the server.

See Chapter 5, "Securing Your Site Against Intruders," for more information about Internet and Windows NT security.

Using Other Windows NT Tools

In addition to Internet Service Manager you can use other tools to configure, control, and monitor the Internet Information services. This sections explains other Windows NT utilities that directly affect your Internet Information Server, and explains how you can use other Windows NT utilities to monitor or enhance your Internet Information Server site.

Configuring Server Options with Control Panel

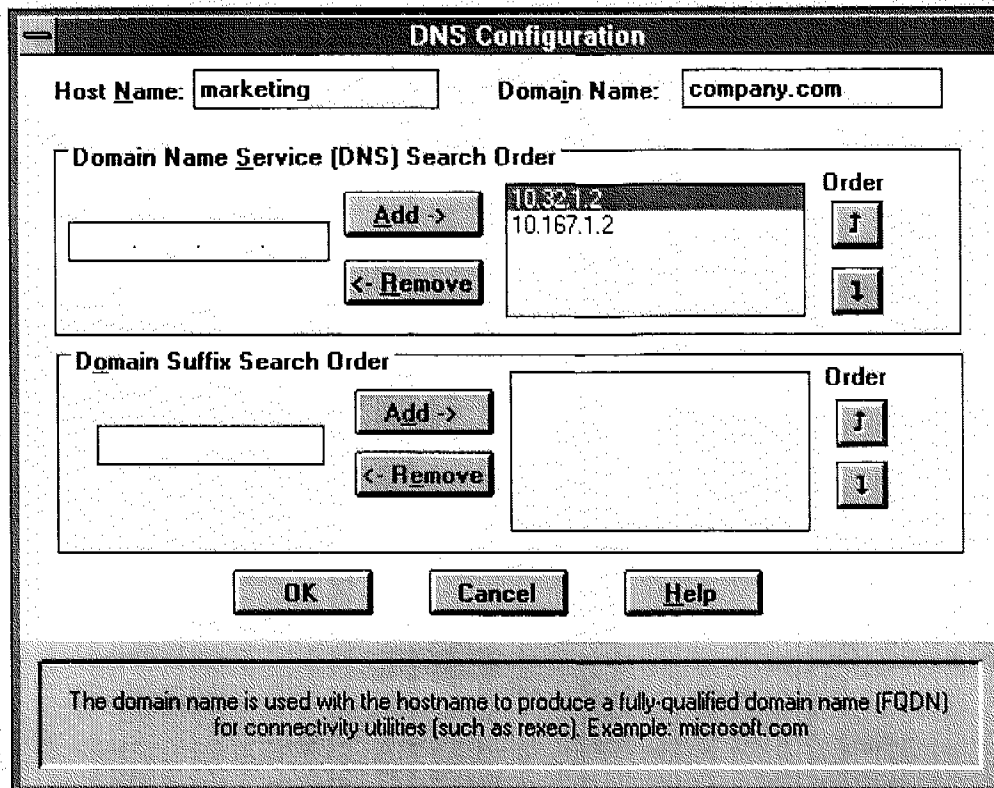
Use Control Panel to set Windows NT-controlled systems and options.

The Network Applet

The Network applet in Control Panel configures your Transmission Control Protocol/Internet Protocol (TCP/IP) settings, including IP address, subnet mask, and default gateway. Doubleclick TCP/IP Protocol in the Installed Network Software listing to display the TCP/IP Configuration dialog box.

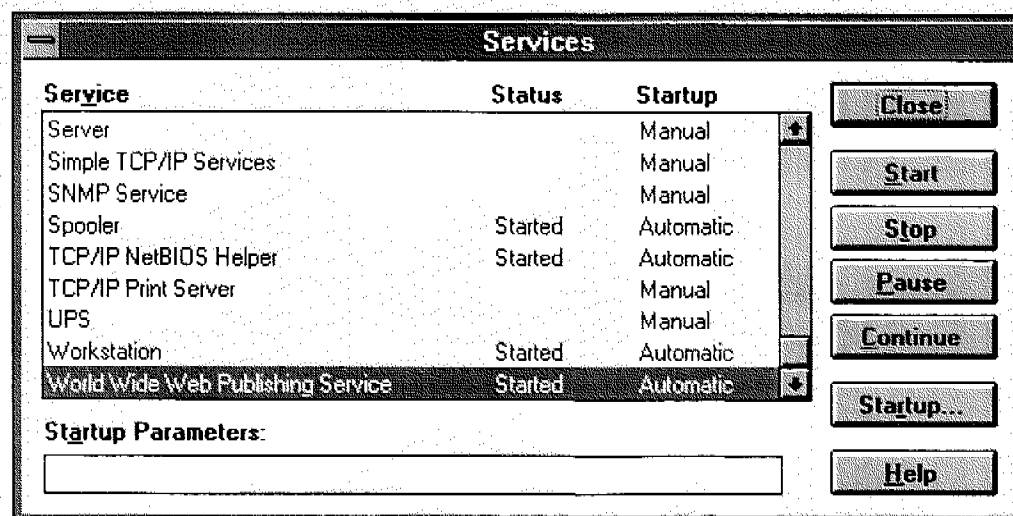
Click the advanced button to set Domain Name System (DNS) settings, such as hostname, domain names,

and DNS servers, to resolve names.

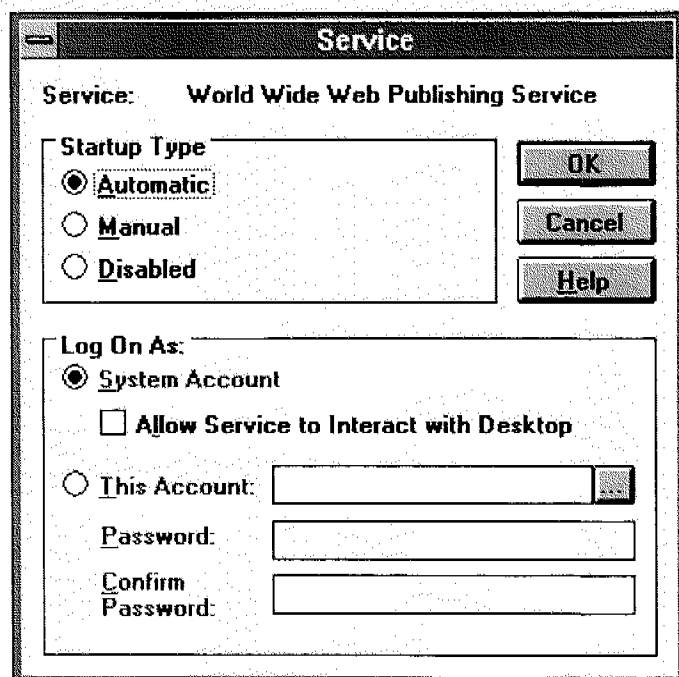


The Services Applet

The Services applet is used to start, stop, and pause the WWW, Gopher, and FTP services. You can also use Internet Service Manager to start, stop and pause the services.



Use the Startup button to configure how the service starts when the computer starts. If you have a specific reason, you can also use this dialog box to override the account used by the WWW service as set in the Service property sheet of Internet Service Manager. You should change this setting only if it is part of your security strategy; otherwise, use the default settings in the Log On As box.

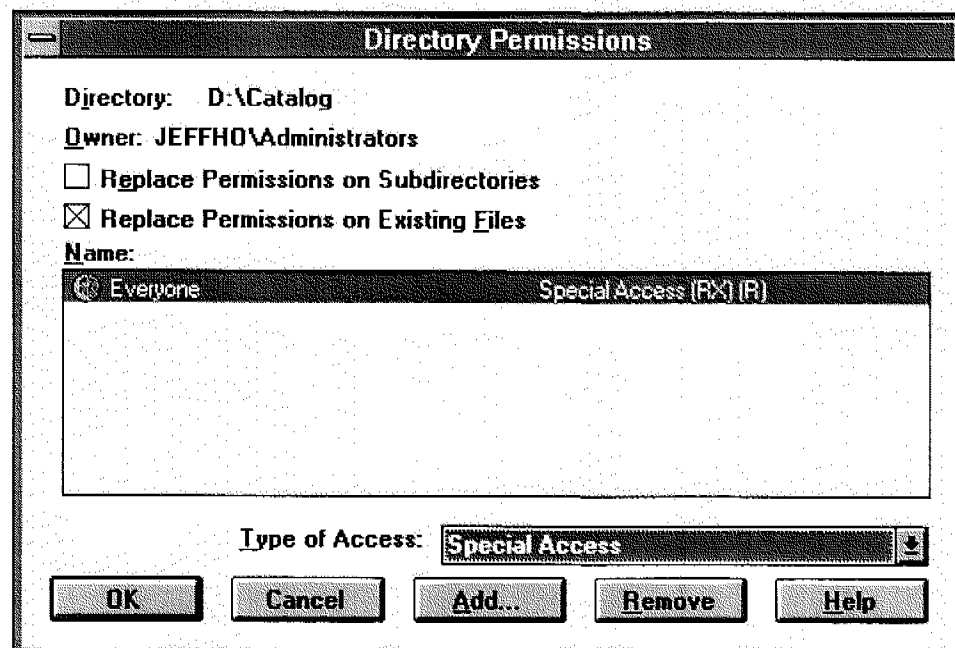


The ODBC Applet

The ODBC applet in Control Panel is used to set up ODBC connectivity. See Chapter 8, "Publishing Information and Applications," for more information about using the ODBC applet.

Setting File Access with File Manager

Use File Manager to set directory and file permissions on Windows NT File System (NTFS) drives. Use the Permissions item in the Security menu to set permissions.



Setting User Access with User Manager for Domains

User Manager for Domains, in the Administrative Tools program group, is a tool that you can use to manage

security for a Windows NT Server computer. With User Manager you can:

- Create and manage user accounts.
- Create and manage groups.
- Manage the security policies.

Tracking Problems with Event Viewer

Event Viewer, in the Administrative Tools program group, is a tool that you can use to monitor events in your system. You can use Event Viewer to view and manage System, Security, and Application event logs. Event Viewer can notify administrators of critical events by displaying popup messages, or by adding event information to log files. The information allows you to better understand the sequence and types of events that led up to a particular state or situation.

Monitoring Your Server with Performance Monitor

Internet Information Server automatically installs Windows NT Performance Monitor counters. With the HTTP service and Internet Services Performance Summary objects, you can use the Windows NT Performance Monitor for real-time measurement of your Internet service use. Similar counters for the Gopher and FTP services are also available.

The WWW service object provides counters to monitor the WWW service; these include:

- Bytes Sent/sec
- Bytes Total/sec
- CGI Requests
- Connection Attempts
- Connections/sec
- Current Anonymous Users
- Current ISAPI Requests
- Current CGI Requests
- Current Connections
- Current NonAnonymous Users
- Files Received
- Files Sent
- Files Total
- Get Requests
- Head Requests
- Logon Attempts
- Maximum Anonymous Users
- Maximum ISAPI Requests
- Maximum CGI Requests
- Maximum Connections
- Maximum NonAnonymous Users
- Not Found Errors
- Other Request Methods
- Post Requests
- Total Anonymous Users
- Total NonAnonymous Users

The Internet Services Performance Summary provides general use and cache-use information about the Internet Information Server; this includes:

- Cache Flushes

- Cache Hits
- Cache Hits %
- Cache Misses
- Cache Size
- Cache Used
- Cached File Handles
- Current Blocked Async I/O Requests
- Directory Listings
- Measured Async I/O Bandwidth usage
- Objects
- Total Allowed Async I/O Requests
- Total Blocked Async I/O Requests
- Total Rejected Async I/O Requests

Last updated January 12, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.



[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#) |

Chapter 5 - Configuring the WWW Service

Lesson 1 HTTP Defined

Lesson 2 WWW Properties

Lesson 3 Virtual Directories

Lesson 4 Virtual Servers

Review

About This Chapter

This chapter describes the basic functionality of Hypertext Transport Protocol (HTTP) 1.1 as it relates to the Microsoft Internet Information Server (IIS) 4.0 World Wide Web (WWW) service. It introduces you to the three different types of property sheets within Internet Information Server and how to access them in order to configure your Web sites. This chapter also describes virtual directories and servers, explaining the various methods by which you can add virtual directories and servers to Internet Information Server.

Before You Begin

To complete the lessons in this chapter, you must have installed Internet Information Server as described in Chapter 2, "Installing Microsoft Internet Information Server."

Lesson 1: HTTP Defined

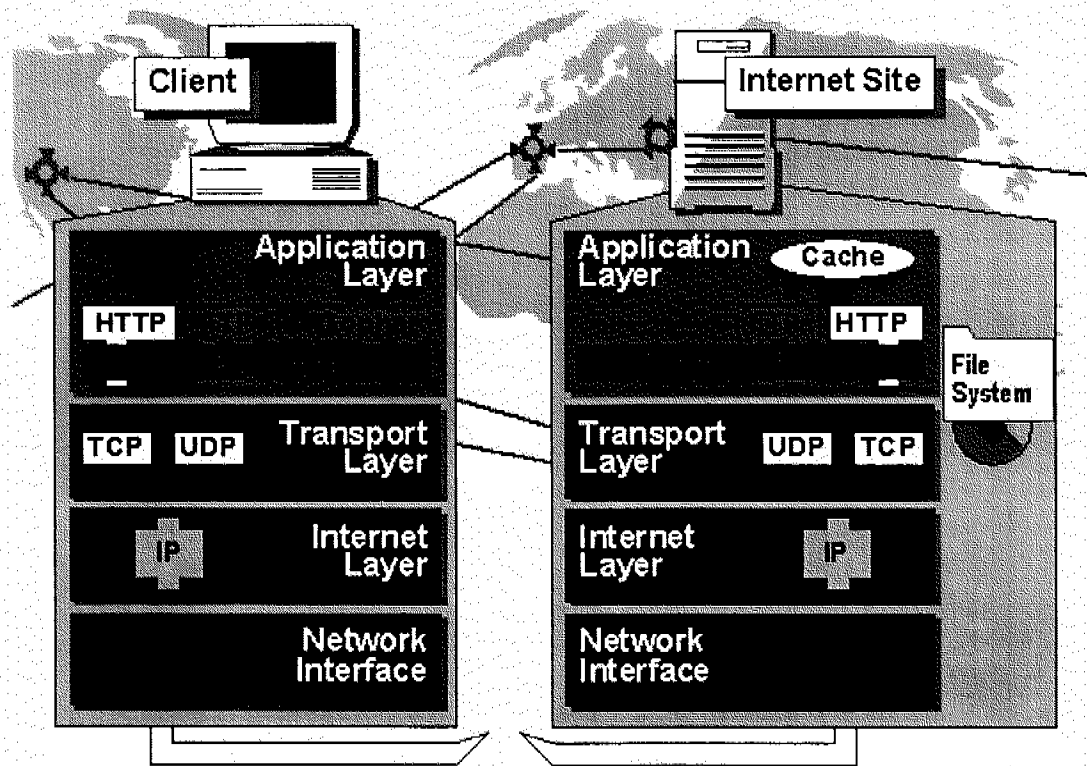
Hypertext Transfer Protocol (HTTP) is the most frequently used protocol on the Internet today. This lesson describes the fundamentals of HTTP.

After this lesson, you will be able to:

- Define HTTP and describe its basic functionality as it relates to the WWW Service.

Estimated lesson time: 5 minutes

HTTP is the protocol that led to the development of the Web. It is a generic, stateless, object-oriented protocol that grew out of a need for a universal protocol to simplify the way users access information on the Internet. HTTP is a client/server protocol located in the Application layer of the Internet protocol stack.



By extending its request methods, or commands, you can use HTTP for many different functions, including name servers and distributed object management systems. Because HTTP categorizes or types data, systems can be built independently of the data being transferred.

HTTP is constantly being improved. The World Wide Web Consortium (W3C) was founded in 1994 to develop common standards for the Web. You can find more information about the W3C at:

<http://www.w3.org/>

General discussions about HTTP and the applications that use HTTP take place on the following mailing list:

www-talk@w3.org

Summary

HTTP is a client/server protocol that was developed to simplify the way users access information on the Internet. It is a generic, stateless, object-oriented protocol that led to the Web. Because HTTP is constantly changing, the W3C was created to develop standards for the Web.

Lesson 2: WWW Properties

Each Web site that you create on your computer has its own set of property sheets. The general settings, or properties, for a site are displayed in these property sheets and stored in the metabase. In this lesson, you learn about the different types of property sheets.

This lesson describes these different property sheets and demonstrates how to access them in order to set the general properties for a site or file within a site on your computer.

After this lesson, you will be able to:

- List and describe the three types of WWW property sheets and their relationship to one another.

- Locate the WWW property sheets and use them to configure the WWW Service.
- List and describe the functions of the property sheets associated with the WWW Service.

Estimated lesson time: 70 minutes

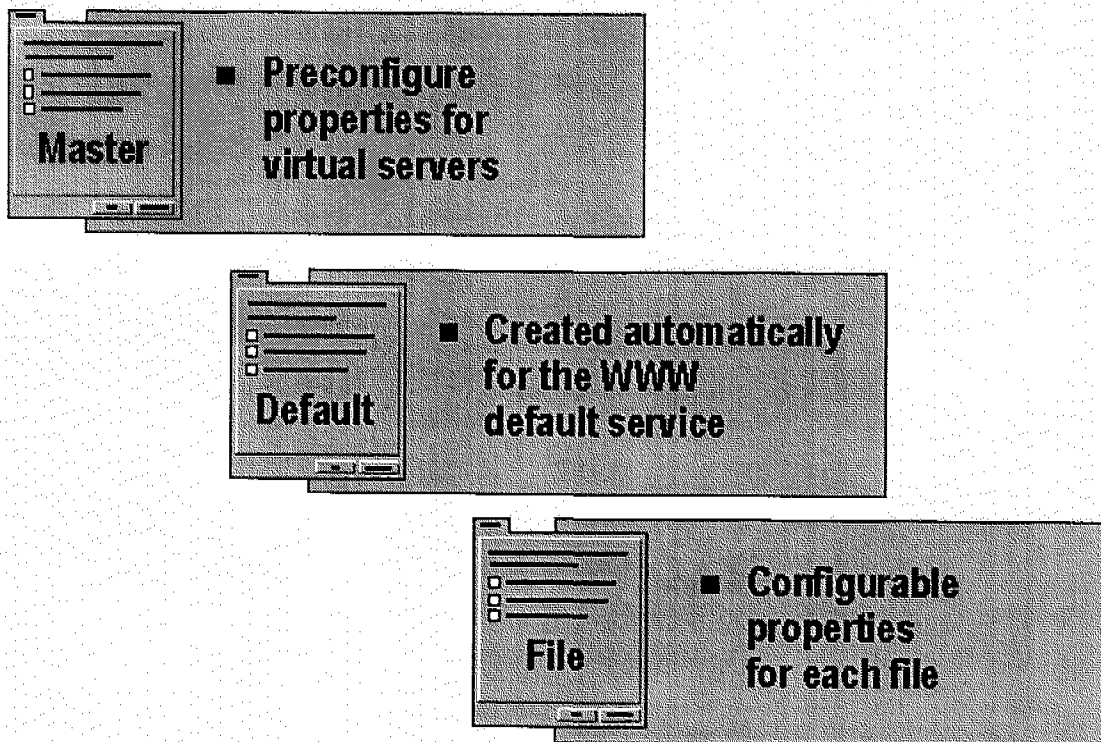
During installation, Internet Information Server assigns default values to the various properties on the different property sheets. You can publish documents on your site without changing these default settings, and you can easily customize the settings as well.

Each Web site that you create and each file within each Web site has an individual set of property sheets that you can edit in order to customize configuration on a file-by-file or site-by-site basis. You can also edit the default property settings so that all subsequent sites or files are created with your custom configuration.

Types of WWW Property Sheets

There are three different types, or classes, of property sheets within Internet Information Server: the **Master**, **Default**, and **File** property sheets. You can customize configuration of all three types of property sheets, but where you make your changes affects the range of influence the changes have on subsequent sites or files created.

It may be helpful to think of the different types of property sheets in terms of a hierarchy with **Master** property sheets at the top of the hierarchy and **File** property sheets at the bottom. **Master** property sheets determine the properties of the virtual Web sites you create, which in turn determine the properties of the files created within each Web site.

**Master**

Master property sheets determine the default properties of every virtual Web site created with this installation of Internet Information Server. During installation, Internet Information Server applies certain default properties to the **Master** property sheets. Every virtual site you create inherits these settings. If you change the settings on the **Master** property sheets, subsequent virtual sites inherit the new settings, but previously created virtual sites do not.

Default

The installation process creates a default Web site with its own default properties. Every file you create within the default Web site inherits these settings.

File

Files created in a virtual directory inherit the virtual directory's property sheet settings, whereas files created in the default Web site inherit the settings of the **Default Web Site Properties** dialog box. After a file is created, the property sheets can be configured on the file level.

WWW Property Sheets

The WWW Service can be configured using a set of nine different property sheets:

- **Web Site**
- **Operators**
- **Performance**
- **ISAPI Filters**
- **Home Directory**
- **Documents**
- **Directory Security**
- **HTTP Headers**
- **Custom Errors**

You can change the settings on these property sheets as needed at any time.

All WWW property sheets can be accessed using the Internet Service Manager (ISM) snap-in for Microsoft Management Console (MMC).

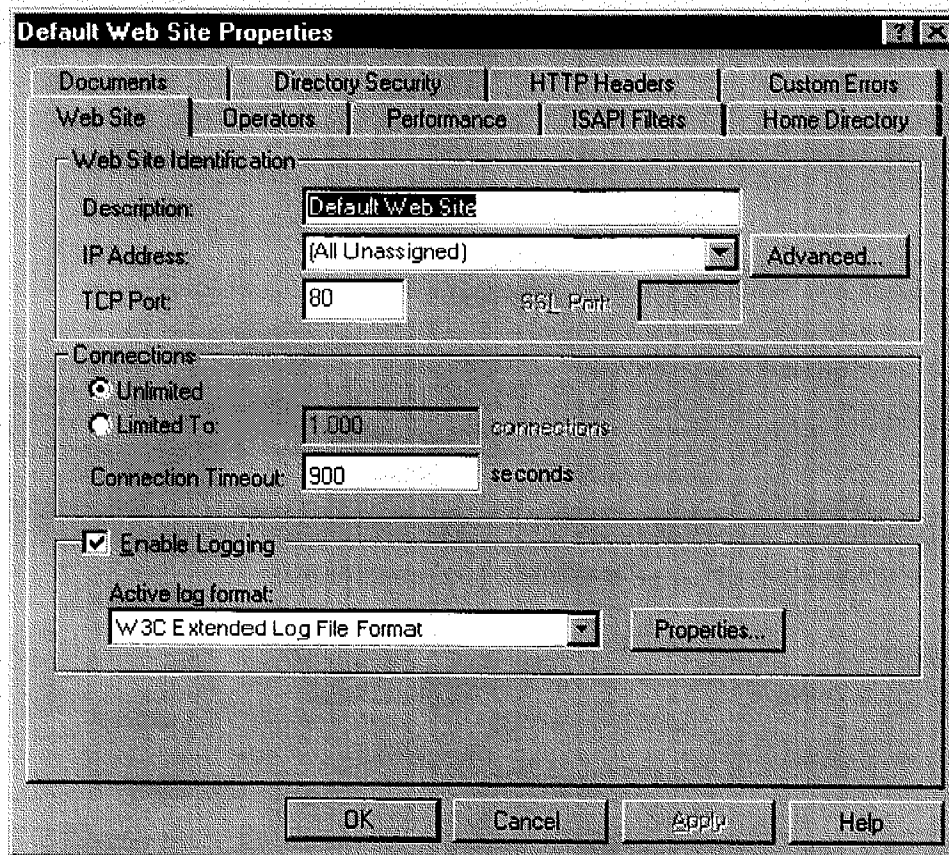
To access the WWW property sheets

1. Click the **Start** button, point to **Programs**, point to **Windows NT 4.0 Option Pack**, point to **Microsoft Internet Information Server**, and then click **Internet Service Manager**.
2. In the left pane, double-click the Internet Information Server node.
3. In the left pane, double-click the *computername* node.
4. Right-click **Default Web Site**, and then click **Properties**

The **Default Web Site Properties** dialog box appears with tabs for each property sheet.

Web Site

You can use the **Web Site** property sheet to set the Web site identification, specify the number of connections allowed, and enable or disable logging for a Web site.



The **Web Site Identification** field allows you to choose a description for your Web site using the following settings:

- **Description.** This dialog box lists the name you choose for your Web site and appears in the tree view of the Internet Service Manager.
- **IP Address.** This dialog box lists the Internet Protocol (IP) address associated with your Web site.
- **TCP Port.** This dialog box determines the port where each service runs. The default is port 80.
- **SSL Port.** This dialog box determines the port used by Secure Sockets Layer (SSL) transmissions.
- **Advanced.** This button opens the **Advanced Multiple Web Site Configuration** dialog box where you can specify additional identities for your Web site.

The **Connections** field allows you to set the number of simultaneous connections to the server using the following settings:

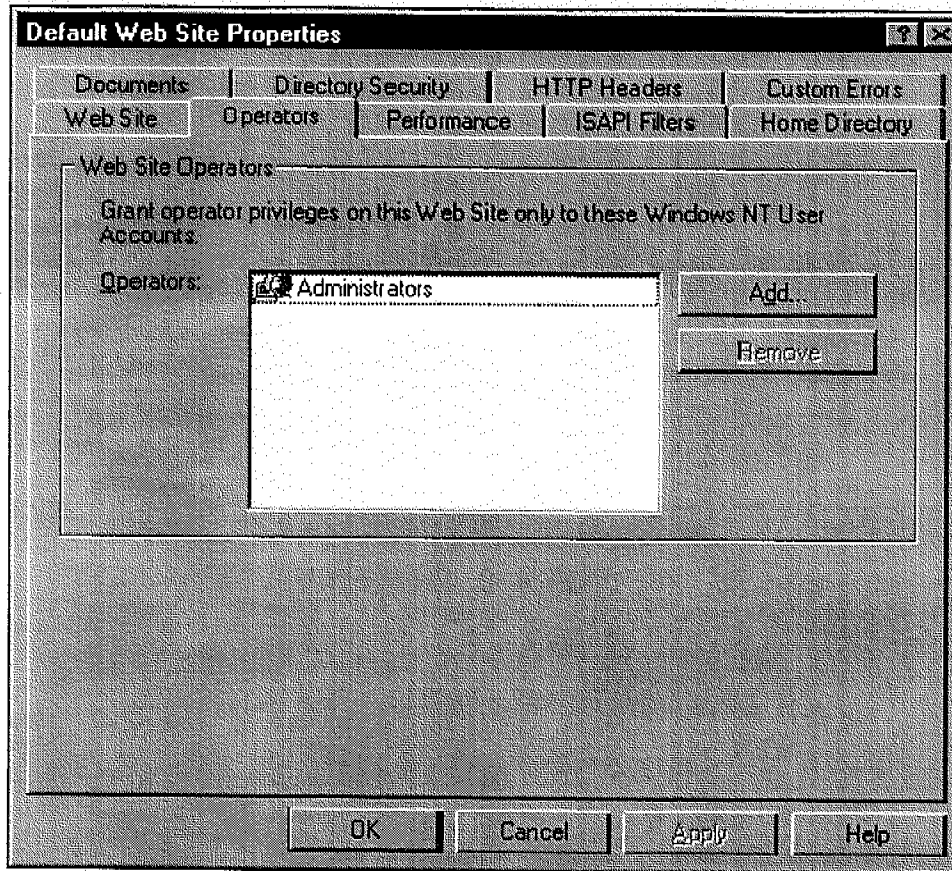
- **Unlimited.** Select this option to allow an unlimited number of simultaneous connections to the server.
- **Limited To.** Select this option to limit the number of simultaneous connections to the server to the number entered in the associated text box.
- **Connection Timeout.** Use this property to set the length of time in seconds before the server disconnects an inactive user.

Select the **Enable Logging** option to activate your Web site's logging features. These can record details about user activity and create logs in your choice of format.

Click the **Properties** button to open the **Microsoft Logging Properties** dialog box. This dialog box allows you to choose how often to create new logs, to specify the file folder for the log, and to select additional extended properties for logging.

Operators

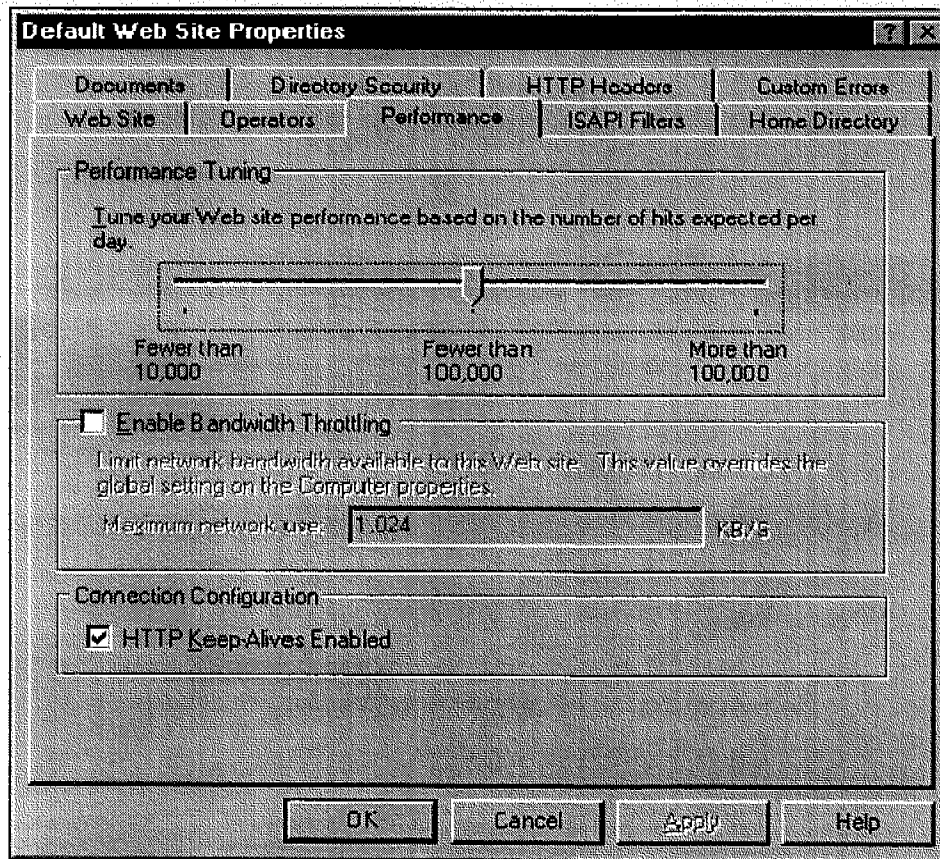
You can use the **Operators** property sheet to control which Microsoft Windows NT User Accounts have administrative privileges for your Web site.



To add a Windows NT User Account to the current list of accounts that have administrative privileges, click the **Add** button. To remove a Windows NT User Account from this list, select the account in the **Operators** box, and then click **Remove**.

Performance

You can use the settings on the **Performance** property sheet to fine-tune your Web site's performance.



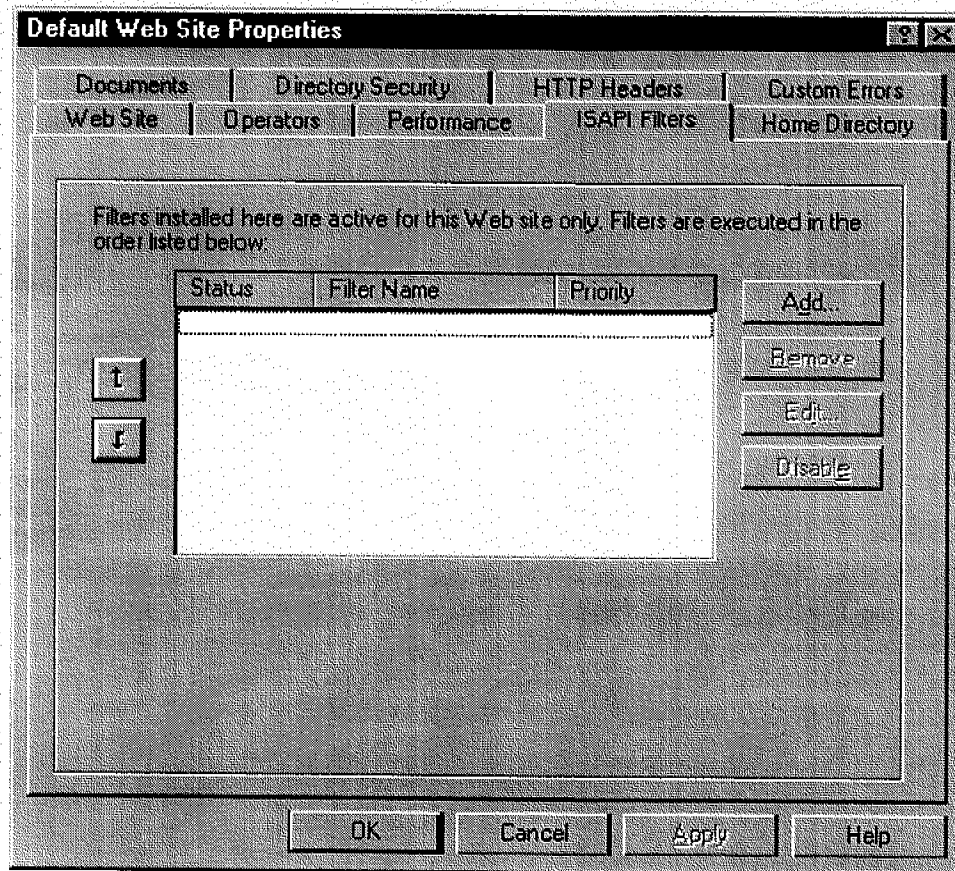
Adjust the **Performance Tuning** setting to the number of daily connections you anticipate for your site. If you set the number slightly higher than the actual number of connections, connections are made faster and server performance is improved. However, if you set it too much higher than the actual number of connection attempts, server memory is wasted and overall server performance is reduced.

The **Enable Bandwidth Throttling** option allows you to limit the bandwidth used by this Web site. For this Web site only—even if it is greater than the value set at the computer level—the bandwidth value entered here overrides the value set at the computer level.

Select the **HTTP Keep-Alive Enabled** box to allow a client to maintain an open connection with your server. This means that the client connection does not have to be reopened with each new request. Keep-Alives are enabled by default.

ISAPI Filters

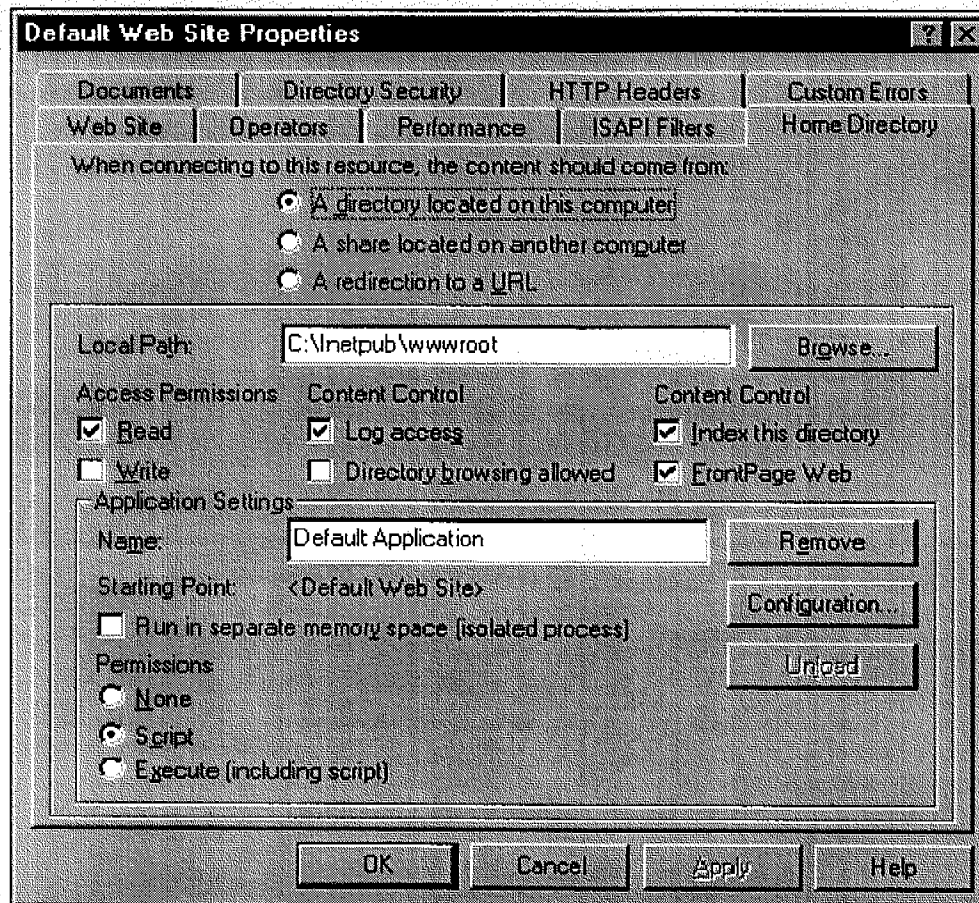
The **ISAPI Filters** property sheet contains options for Internet Server Application Programming Interface (ISAPI) filters. You can use ISAPI to run remote applications. Requesting a Uniform Resource Locator (URL) that is mapped to a filter activates these applications. You can use these settings to map file name extensions to the correct filter on the Web server.



The table on this property sheet lists the status (Loaded, Unloaded, or Disabled), name, and the priority rating set inside the dynamic-link library (DLL), which is High, Medium, or Low, of each filter. You can modify filter mappings with the **Add**, **Remove**, and **Edit** buttons. You can use the **Enable** and **Disable** buttons to modify a filter's status. Select a filter and then click the up or down arrow button to change the order in which the server runs the ISAPI filters.

Home Directory

You can use the **Home Directory** property sheet to change your Web site's home folder and modify its properties.



The home folder is the central location for the files published in your Web site. Installation of the WWW Service creates a default home folder named Wwwroot. You can use the buttons at the top of this property sheet to change the location of the current Web site's home folder to one of the following:

- A folder located on the same computer
- A share located on another computer
- A redirection to a URL

If you change the home folder, type the precise path to the new folder, share, or destination URL in the **Local Path** box, or use the **Browse** button to locate the folder path.

Access Permissions properties are applicable when your home folder is a local folder or a network share. The following check boxes determine the type of access the folder allows:

- **Read.** **Read** access permission enables Web clients to read or download files stored in either the home folder or a virtual directory. You learn more about virtual directories later in this chapter.
- **Write.** **Write** access permission enables Web clients to upload files to the enabled folder on your server, or to change the content of a write-enabled file. However, Web clients can only perform Write-access procedures with a browser that supports the PUT feature of the HTTP 1.1 protocol standard.

The following **Content Control** properties are applicable when your home folder is a local folder or a network share:

- **Log access.** **Log access** allows you to record visits to this folder in a log file.
- **Directory browsing allowed.** By selecting this check box you allow the server to compile a

hypertext listing of the files and subfolders within this folder. This listing is generated automatically and sent to the user whenever a browser request does not include a specific file name or when the server cannot find one of the specified default documents in the folder. For more information, see the discussion of the **Documents** property sheet later in this lesson. This listing allows the user to navigate through the folder structure.

- **Index this directory.** Selecting this check box instructs Microsoft Index Server to include this folder in the full-text index of your Web site.
- **FrontPage Web.** Select this check box to create a Microsoft FrontPage Web site for this folder. FrontPage allows you to manage your Web site, as well as create the site content.

Within the WWW Service, an application is defined as all of the folders and files contained within a folder. The application begins at a point marked as an application starting point and runs until another application starting point is reached. If you make your site's home folder an application starting point, then every virtual directory and physical folder within your site can participate in the application.

To dissociate this home folder from an application, click the **Remove** button. You can make this folder an application starting point (and thus create an application) by clicking the **Create** button. Type the name of the application in the **Name** box, and the name appears in the property sheets for any folder contained within the application boundary.

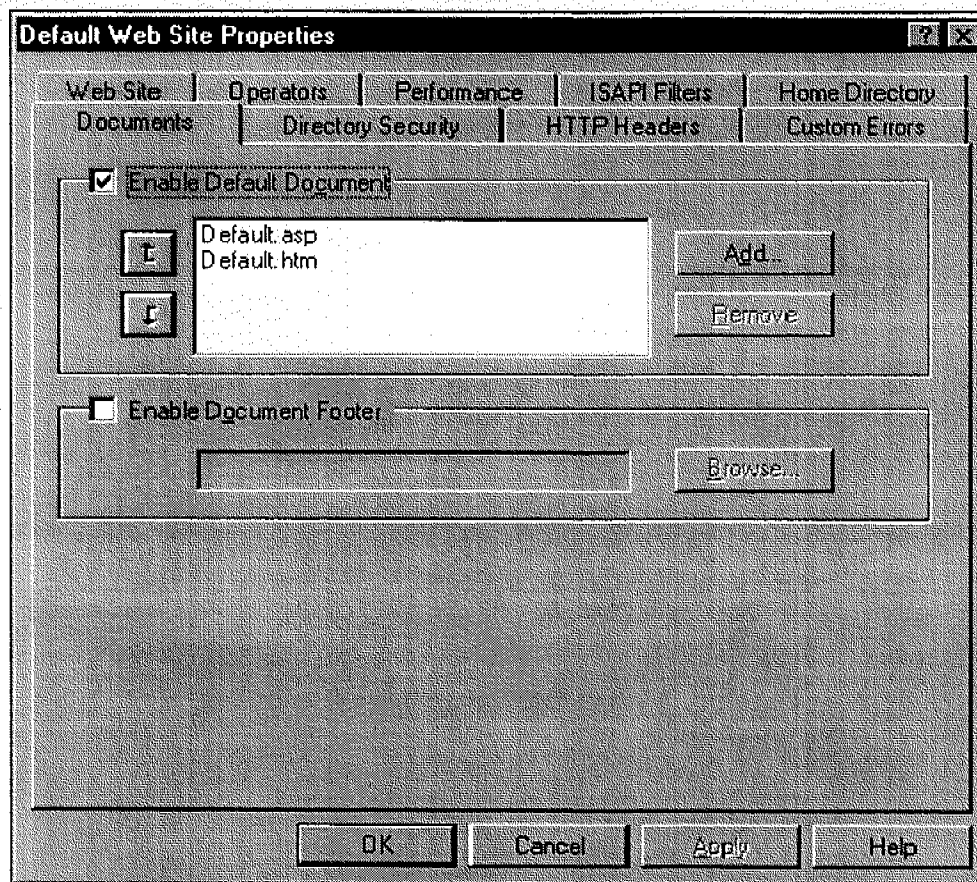
- **Run in separate memory space (isolated process).** Select this check box to run this application in a process separate from the Web server process. Running an isolated application protects other applications, including the Web server itself, from being affected if this application becomes unavailable or stops responding.
- **Permissions.** Use this setting to control whether other applications can be run in this folder. Select **None** if you do not want to allow any programs or scripts to run in this folder. Selecting **Script** enables a script engine to run in this folder without having set Execute permissions. The **Execute (including script)** setting allows any application to run in this folder, including script engines and Windows NT binaries (.dll and .exe files).

Click the **Configuration** button to set application-specific properties. There are four application configuration property sheets:

- **Application Mappings.** Use this page to map file name extensions to the applications that process those files.
- **Active Server Pages (ASP).** Use this page to set the options that control how ASP scripts run.
- **ASP Debugging.** Use this page to set debugging options for ASP scripts.
- **Other.** Use this page to set or change the CGI Script Timeout value.

Documents

You can use the **Documents** property sheet to specify default documents and attach default footers to your Web pages.



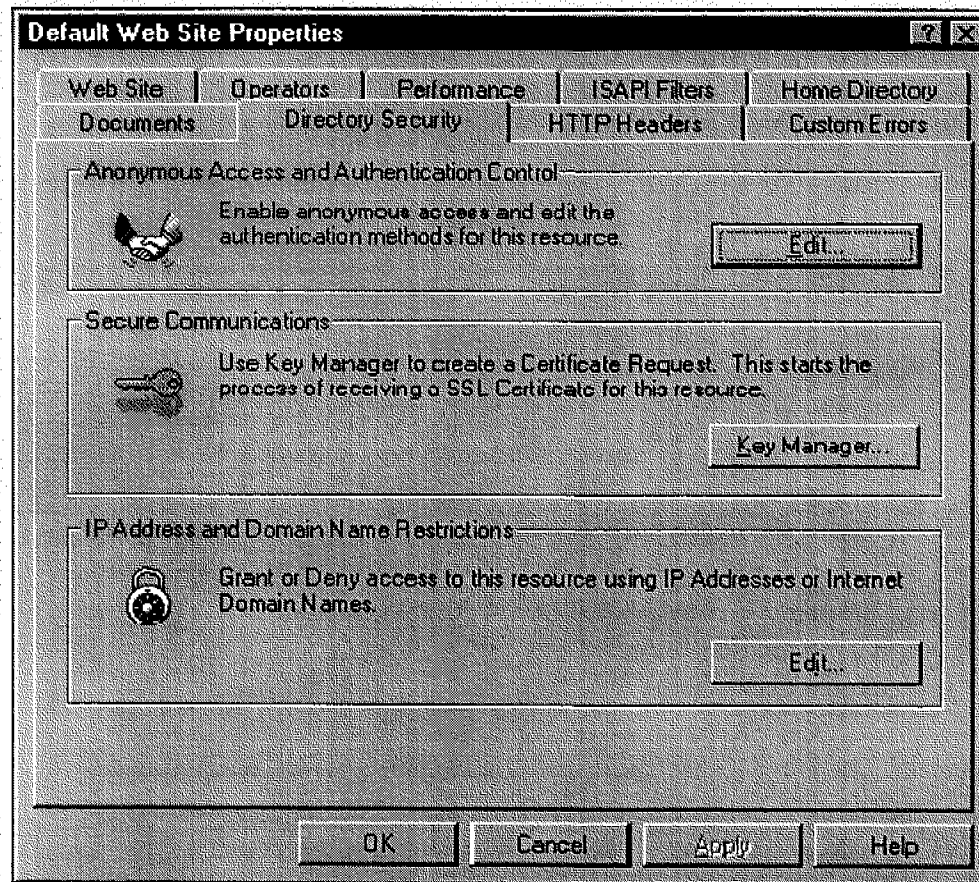
By selecting the **Enable Default Documents** check box, you can show the user a default document when a browser request does not include a specific Hypertext Markup Language (HTML) file name. Default documents can be a folder's home page or an index page that provides links to the documents in the folder. For more information, see the discussion of the **Home Directory** property sheet earlier in this lesson. You can specify more than one default document. To add a new default document, click **Add**.

When prompted by a browser, the Web server searches the folder for default documents, following the order in which the names appear in this list. The server returns the first document it finds. To change the search order, select a document, and then click the up or down arrow button.

Select the **Enable Document Footer** option to configure your Web server to insert a footer automatically. Your footer must be a separate file, but it should not be a complete HTML document. Your footer file should include only the HTML tags necessary for formatting the appearance and function of your footer content. For example, your footer file can contain HTML formatting instructions for adding a logo image and identifying text to your Web pages. You must provide the full path and file name for your footer file.

Directory Security

You can use the **Directory Security** property sheet to configure your Web server's user identification security features.



The **Anonymous Access and Authentication Control** option sets the anonymous access and authentication control methods for access to the server. Click **Edit** to select one or more authentication methods from the following options:

- **Allow Anonymous Access.** Select this box to allow anonymous users to log on to your Web server. When a user establishes an anonymous connection, your server logs the user on with an anonymous or guest account. In either case, the account used is a valid Windows NT user account. Click **Edit** to specify which Windows NT User Account to use for anonymous connections.
- **Basic Authentication.** Select this box to enable your Web server's Basic Authentication method, where the password is sent in clear text. By selecting this option, a user name and password are required when the **Allow Anonymous Access** option is disabled, or access to the server is controlled using Windows NT File System (NTFS) access control lists.
- **Windows NT Challenge/Response.** Select this box to enable your Web server's Windows NT Challenge/Response authentication methods. As with Basic Authentication, a user name and password are required when the **Allow Anonymous Access** option is disabled, or access to the server is controlled using NTFS access control lists. Windows NT Challenge/Response is supported by Microsoft Internet Explorer, version 2.0 or later.

For a discussion of authentication and password security issues, see Chapter 9, "Adding Windows NT and Internet Information Server Security Features."

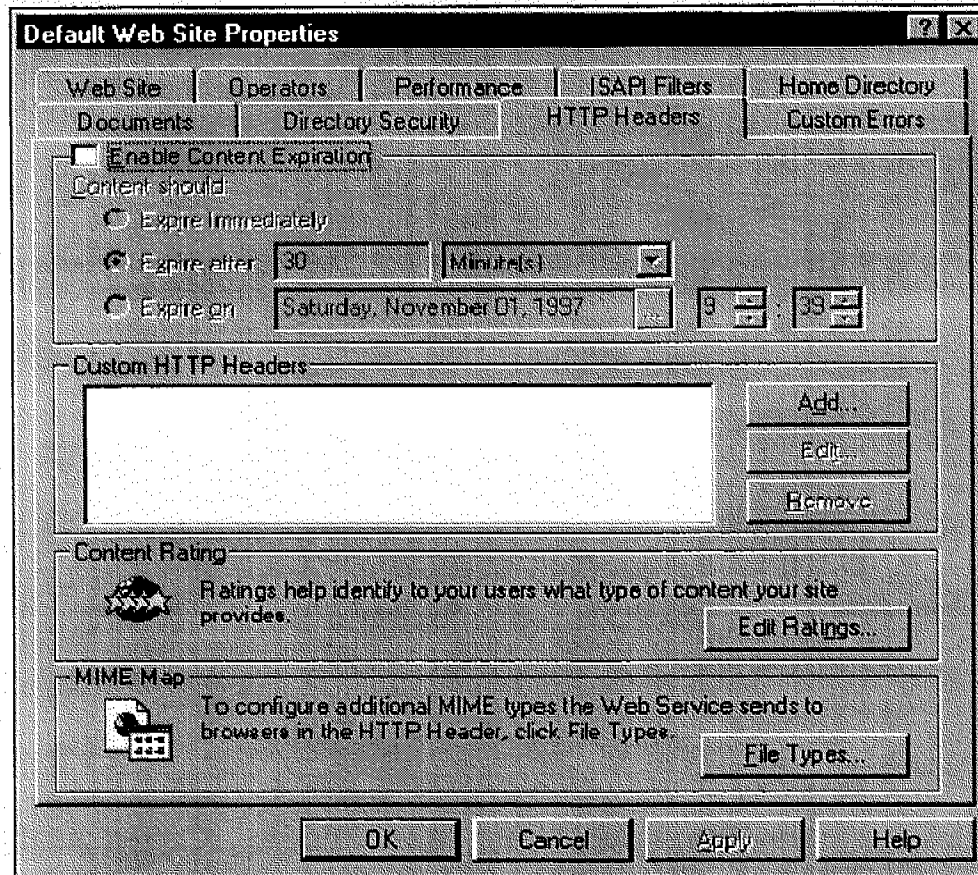
The **Secure Communications** feature is available only for Windows NT Server installations. This feature uses Key Manager to create a certificate request. Click **Key Manager** to start the process of receiving a SSL digital certificate for this resource.

Use the **IP Address and Domain Name Restrictions** properties to grant or deny access to this resource using IP addresses or Internet domain names. Click **Edit** to grant or deny access to specific individuals or groups as follows:

- **Granted Access.** Click this button to grant access to all computers by default. Click **Add** to list those computers that are denied access.
- **Denied Access.** Click this button to deny access to all computers by default. Click **Add** to list those computers that are granted access.

HTTP Headers

You can use the **HTTP Headers** property sheet to set values returned to the browser in the header of specific HTML pages.



Select the **Enable Content Expiration** check box to include expiration information in the HTML page header. When you include a date in time-sensitive material, such as special offers or event announcements, the browser compares the current date to the expiration date and determines whether to display a cached page or request an updated page from the server.

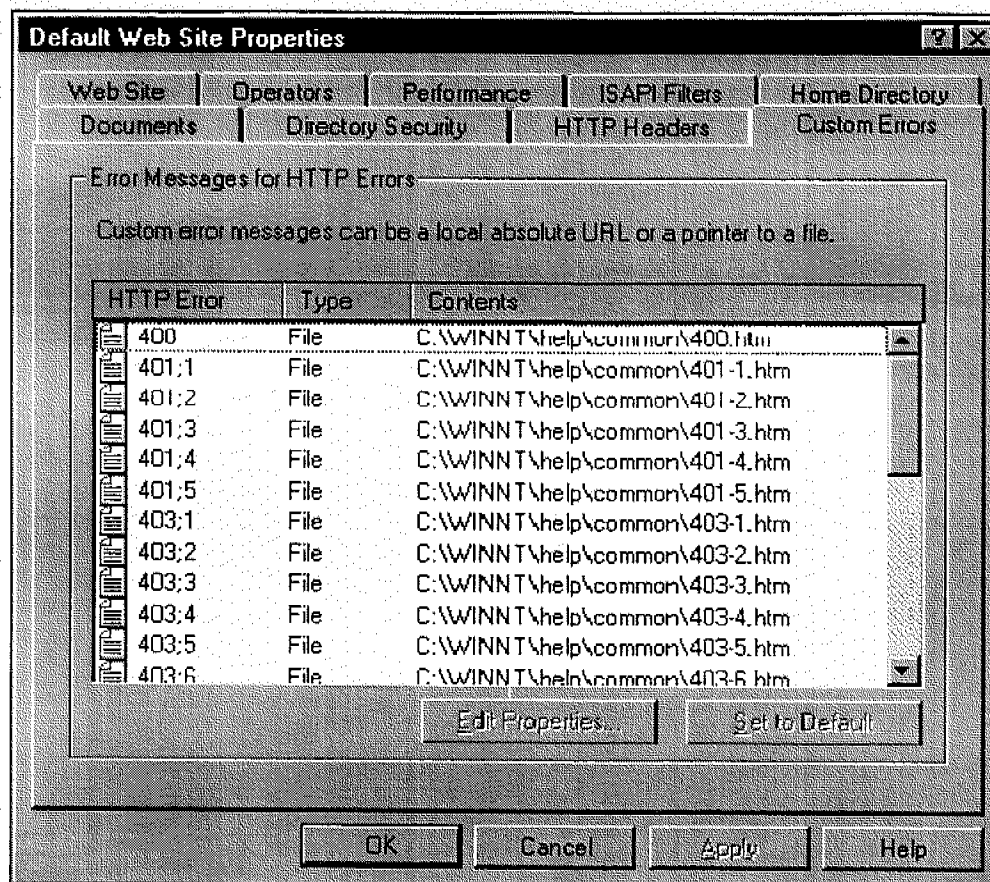
You can send custom HTTP headers from your Web server to the client browser. To send a header, click **Add**, and then type the name and value of the header in the **Add Custom HTTP Header** dialog box. Click **Remove** to stop sending the header.

You can embed descriptive labels in your Web page's HTTP headers using the **Content Rating** feature. Some Web browsers, such as Microsoft Internet Explorer version 3.0 or later, can detect these content labels in order to help users identify potentially objectionable Web content. Click **Edit Ratings** to set content ratings for this Web site, folder, or file.

You can set which file types your Web service returns to browsers. Clicking the **File Types** button allows you to configure Multipurpose Internet Mail Extensions (MIME) mappings.

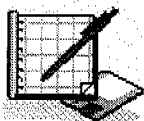
Custom Errors

The **Custom Errors** property sheet lists the messages returned to the browser in the event of an HTTP error.



You can use either the default HTTP 1.1 errors or you can customize these error messages with your own content. When error messages are customized, the HTTP error code is still listed, as well as the output type, which can be the default HTTP 1.1 error, a local absolute URL, or a pointer to a file.

Practice



In this practice scenario, you use Internet Service Manager to take your Web site offline without stopping the WWW Service. You then create a custom error message, and replace the default error message with your custom error message. Finally, you use Internet Service Manager to bring your Web site online without restarting the WWW Service. You can use the procedures in this practice if you need to inform visitors that your site is offline due to maintenance.

In the first part of the practice, you take your default Web site offline.

To take your default Web site offline

1. Start Microsoft Management Console with the Internet Service Manager snapin.
2. In the left pane, double-click the Internet Information Server folder.

The Internet Information Server folder opens displaying a computer icon.

3. In the left pane, double-click the computer icon.

The computer tree expands displaying the default sites.

4. Right-click **Default Web Site**, and then click **Properties**.

The **Default Web Site Properties** dialog box appears displaying the **Web Site** tab.

5. Click the **Home Directory** tab.
6. Under **Access Permissions**, click **Read**.

The **Read** check box is cleared.

7. Click **OK**.

The **Inheritance Overrides** dialog box appears.

8. Click **Select All**.
9. Click **OK**.

You now test your Web site for the standard error message.

To test your Web site with Internet Explorer

1. Open Internet Explorer.
2. In the **Address** box, type your server name.
3. Click **OK**.

The message

HTTP Error 403 Access Forbidden
403.2 Forbidden: Read Access Forbidden

appears with a paragraph describing the error beneath it.

4. Minimize Internet Explorer.

Next, you create a custom error message to replace the standard error you viewed in the previous practice.

To create the custom error message by editing the existing message

1. Open Notepad.
2. On the **File** menu, click **Open**.

The **File name** box appears.

3. Type **c:\winnt\help\common\403-2.htm**
4. Click **Open**.

Notepad displays the HTML code for error 403. The main error message text is in the middle of the screen. It begins "This error canU" and is bounded by a paragraph beginning marker, <p>, and a paragraph ending marker, </p>.

5. Replace the error message text with: "Our site is closed for repairs. Please try again later."
6. On the **File** menu, click **Save As**.
7. The **Save As** dialog box appears.
8. Save the file in the C:\Winnt\Help\Common folder.
9. In the **File Name** box, type **err403.htm**

10. Click **Save**.
11. Close Notepad.

In this practice, you install the custom error message on your Web site.

Note This step is necessary because you are redirecting the Help engine to a different file (Err403.htm) when encountering Error 403.2 rather than permanently replacing the standard file. This way, you can easily direct the Help engine back to the standard message when you no longer need the custom message.

If you had saved the custom message over the standard file, this step would not be necessary.

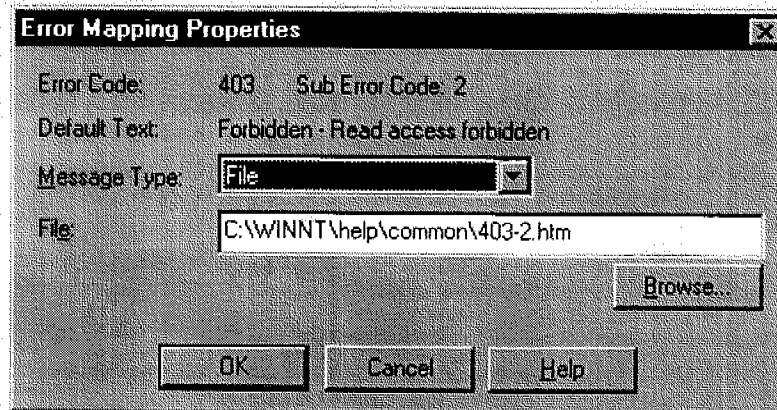
To install the custom error message

1. Start Internet Service Manager.
2. Right-click **Default Web Site**, and then click **Properties**.

The **Default Web Site Properties** dialog box appears.

3. Click the **Custom Errors** tab.
4. Select the 403;2 HTTP error.
5. Click **Edit Properties**.

The **Error Mapping Properties** dialog box appears.



6. In the **Message Type** box, select **File**.
7. In the **File** box, type **c:\winnt\help\common\err403.htm**
8. Click **OK** to return to the **Default Web Site Properties** dialog box.
9. Click **OK**.

The **Inheritance Overrides** dialog box appears.

10. Click **Select All**.

In the **Child Nodes** text box, **IISADMIN** and **IISHELP** are selected.

11. Click **OK**.

You now test your Web site for the custom error message.

To test your Web site with Internet Explorer

1. Switch to Internet Explorer.
2. Click **Refresh**.

The message

HTTP Error 403 Access Forbidden
403.2 Forbidden: Read Access Forbidden

appears with your new error text beneath it.

3. Close Internet Explorer.

To complete the practice, you bring your default Web site online.

To bring your default Web site online

1. Switch to Internet Service Manager.
2. Right-click **Default Web Site**, and then click **Properties**

The **Default Web Site Properties** dialog box appears displaying the **Web Site** tab.

3. Click the **Home Directory** tab.
4. Under **Access Permissions**, click **Read**

The **Read** check box is selected.

5. Click **OK**.
6. Close Microsoft Management Console.

You are prompted to save the changes to Iis.mcs.

7. Click **No**.

Microsoft Management Console closes without saving your changes.

Summary

There are three different types, or classes, of property sheets within Internet Information Server. They are the **Master**, **Default**, and **File** property sheets. The property sheets in Internet Information Server are organized within a hierarchy such that the settings of the **Master** property sheets are passed on to the **Default** property sheets, which are passed on to the **File** property sheets.

You can use WWW property sheets to configure the different areas of your Web site, folder, or file. Each set of the WWW property sheets is made up of the following nine component property sheets:

- **Web Site**
- **Operators**
- **Performance**
- **ISAPI Filters**
- **Home Directory**
- **Documents**
- **Directory Security**
- **HTTP Headers**
- **Custom Errors**

You can change the settings on these property sheets at any time as needed.

Lesson 3: Virtual Directories

A virtual directory is a folder that is not physically contained within the Internet Information Server service (WWW or File Transfer Protocol [FTP]) home folder, but which appears as though it were to users who visit your Web site. In this lesson, you learn about the different types of virtual directories, how to create them, and how to administer them.

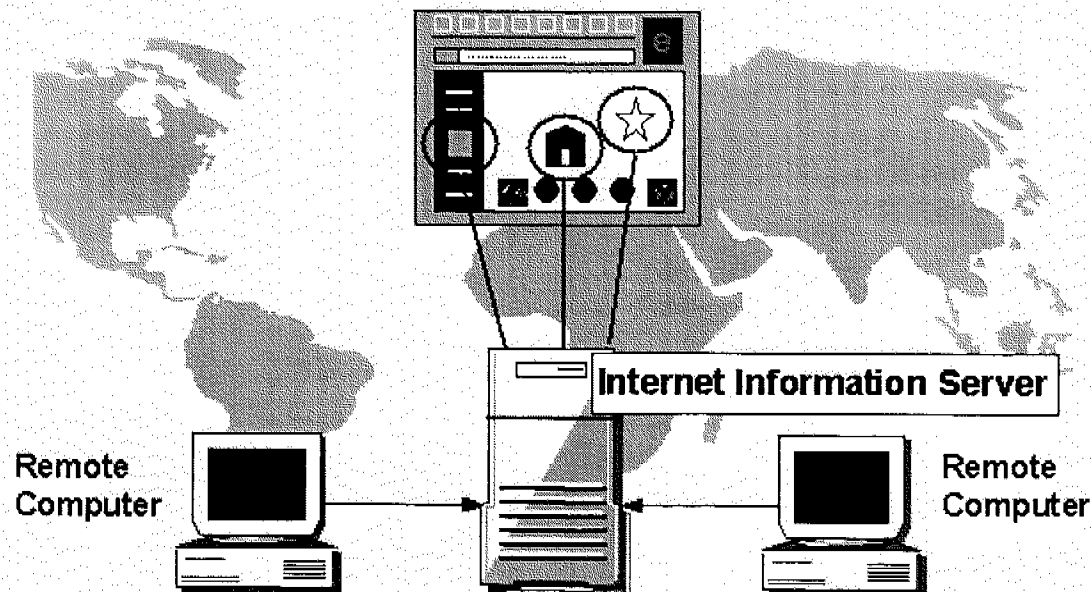
After this lesson, you will be able to:

- Describe the different aspects of a virtual directory.
- Create and administer a WWW Service virtual directory using the ISM snapin.

Estimated lesson time: 20 minutes

Virtual directories increase your flexibility when determining where to store files on your server. By using virtual directories you can store files where they are most easily updated or accessed. Virtual directories also allow you to add storage capacity for your Web site without having to shut down your server.

However, you may experience a drop in performance when accessing folders contained on another computer's disk. This performance drop is due to the transfer speed of data over a LAN.



Virtual directories can be established for both WWW and FTP Services running on Internet Information Server. Virtual directories can be created for folders located on:

- The same disk as the Wwwroot or Ftproot (home) directories.
- Another disk inside the local computer.
- Another computer's disk on the network. This computer must be located within the same Windows NT Server domain as the Internet Information Server computer.

Local Virtual Directory

You can create local virtual directories for folders stored on any disk installed in the same computer as the disk running Internet Information Server.

When configuring a local virtual directory, you must assign an alias to the folder. This alias can be the folder's name or any other name that identifies the site to the user. You must also be prepared to provide the virtual directory's full path.

Remote Virtual Directory

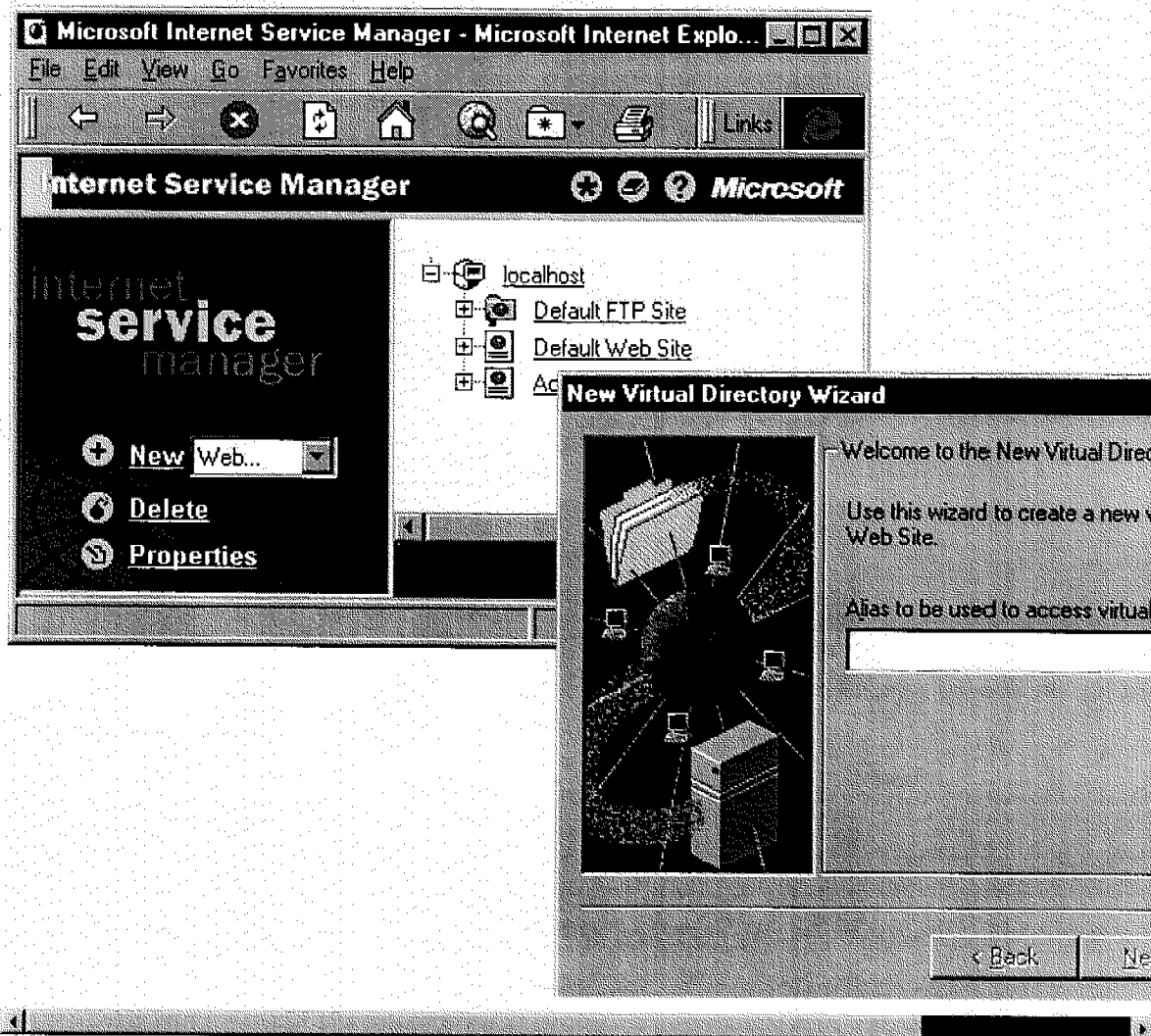
You can use remote virtual directories for folders stored on disks installed in other computers within the Internet Information Server computer's domain.

As with local virtual directories, when configuring a remote virtual directory, you are asked to assign an alias to the folder. You must also supply the folder's universal naming convention (UNC) address. In order to access the folder with a UNC, you have to enter a valid user name and password. The user name and password you enter is automatically used by visitors who access data contained within this virtual directory.

Caution Make sure that the user account you establish to allow Internet access to a remote virtual directory provides only the minimum permissions required to use the site. Do not use the administrator's account to access virtual directories.

Virtual Directory Administration

You can create a virtual directory with any Internet Information Server administration tool. Each tool uses a different user interface for creating virtual directories.



The ISM snap-in for Microsoft Management Console uses the New Virtual Directory wizard to lead you through the virtual directory creation process. After you have established the virtual directory, you can use the **Virtual Directory** property sheet to modify its configuration.

HTML-based Administration (HTMLA), the HTML-based ISM, uses a Web page to lead you through the creation of the virtual directory. You can use this administration tool to establish and modify virtual directories remotely.

With Windows Scripting Host (WSH) you can create virtual directories automatically using scripts.

Practice



In this practice, you set up a virtual directory. To set up a virtual directory, you must use Microsoft Windows NT Explorer to create and share the folder with the appropriate permissions. You then use ISM to create the virtual directory.

To create a local virtual directory

1. Create the C:\Inetpub\Vdir folder.
2. Copy the Default.asp file from the C:\Inetpub\Wwwroot folder to the C:\Inetpub\Vdir folder.
3. Start Internet Service Manager.
4. Right-click **Default Web Site**.

A context menu appears.

5. Click **New**, and then click **Virtual Directory**.

The New Virtual Directory wizard appears.

6. Type **MyVirSite** as the alias to be used.
7. Click **Next**.
8. Type the path **c:\inetpub\vdir** and then click **Next**.
9. Click **Finish**.
10. Start Internet Explorer and then in the **Address** box, type **http://computename/myvirsite**

The Internet Information Server home page appears.

11. Switch to Internet Service Manager.
12. In the left pane, select **MyVirSite**, and then press **F2**.

The ISM interface allows you to rename **MyVirSite**.

13. Rename **MyVirSite** to **OldVirSite**.
14. Switch to Internet Explorer, and then click **Refresh**.

The HTTP/1.0 404 Object not found message appears.

15. In the **Address** box, type **computename/oldvirsite** and then press ENTER.

The Internet Information Server home page appears.

Virtual Directories and FrontPage

The Microsoft FrontPage Web authoring and management tool automatically manages virtual directory use. When installed, FrontPage sets up virtual directories for the folders containing executable FrontPage server extensions. In addition, by marking folders as executable you permit them to include such executable objects as:

- Active Server Pages (.asp)
- Internet database connector files (.idc)
- Common Gateway Interface (CGI) scripts (.exe)
- ISAPI extensions (.dll)
- Practical Extraction and Report Language (PERL) scripts (.pl)

Note Because it does not support noncontiguous content areas, you cannot use virtual directories to merge noncontiguous content areas in FrontPage.

Summary

A virtual directory is a folder that is not physically contained within the Internet Information Server service (WWW or FTP) home folder. You can create two types of virtual directories with Internet Information Server, local and remote. A local virtual directory is located on a disk contained within the same computer as the disk running Internet Information Server. A remote virtual directory is located on a different computer located within the same domain as the computer running Internet Information Server. You can create virtual directories with any Internet Information Server administration tool, and use Microsoft FrontPage to automatically manage them.

Lesson 4: Virtual Servers

Multiple domain names can be hosted on a single computer running Internet Information Server by using virtual servers. This lesson describes virtual servers, their setup, and their administration.

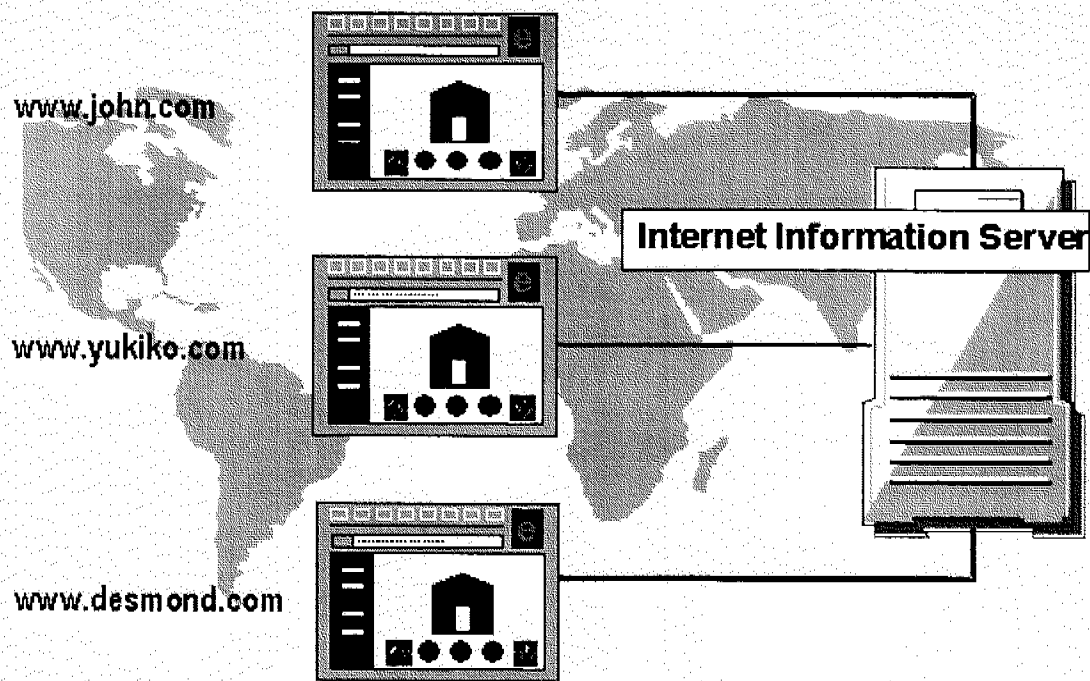
After this lesson, you will be able to:

- Describe two methods of assigning an IP address to virtual servers running on Internet Information Server.
- Create and administer a WWW Service virtual server.

Estimated lesson time: 15 minutes

With virtual servers you can host multiple Web and FTP sites on a single computer running Internet Information Server, which means you do not need to allocate one computer and software package for each site. You simply need to obtain a unique IP address for each domain name assigned to the server, and by using Host Headers you can use a single IP address for multiple domain names. However, only WWW sites can use Host Headers. Virtual servers also centralize administration and simplify server software upgrades.

Hosting multiple virtual servers on the same computer may reduce overall server performance, and virtual Web servers using Host Headers require an HTTP version 1.1-compliant Web browser.



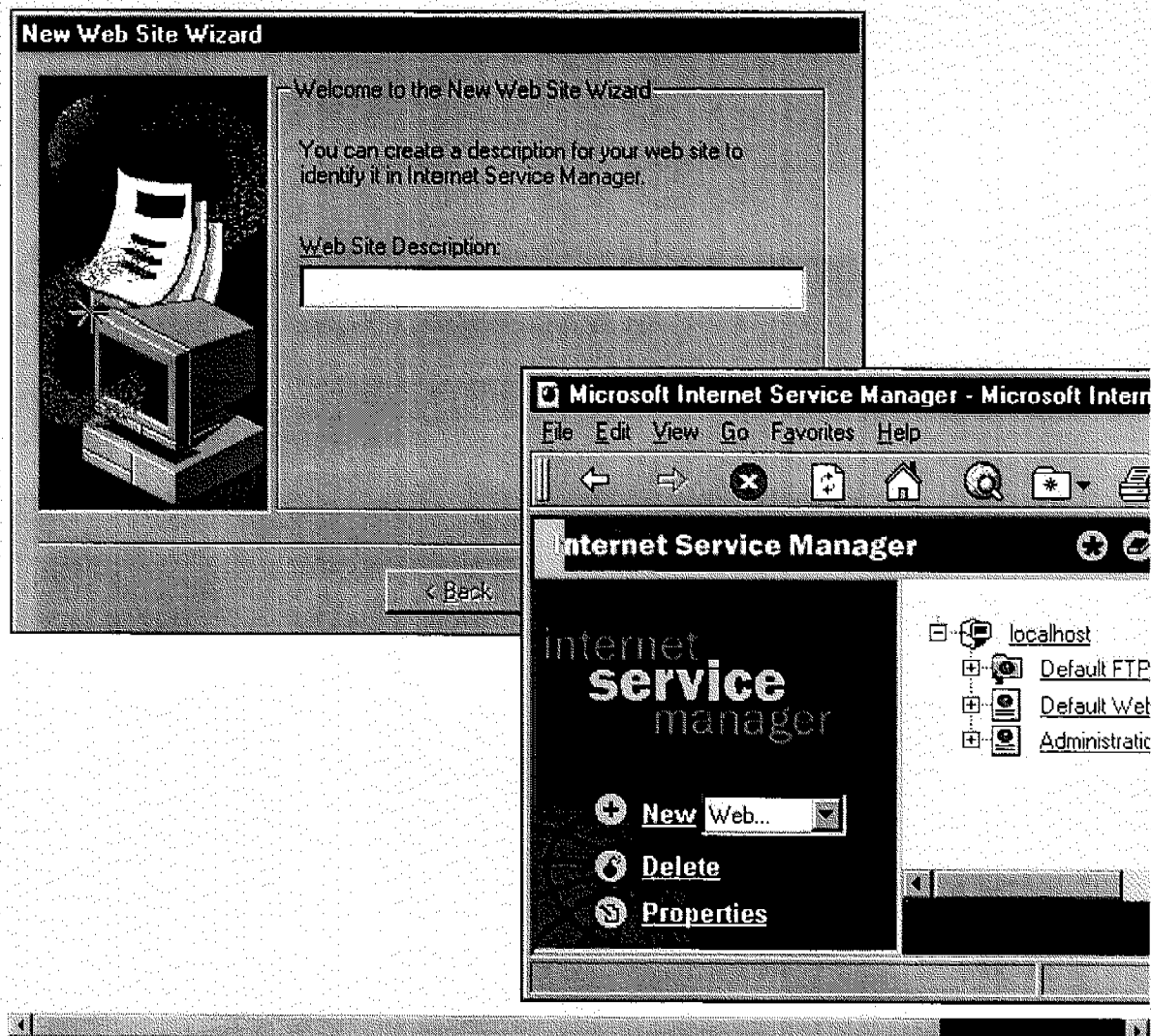
Host Headers

The Host Headers capability of HTTP 1.1 allows you to associate multiple host names with a single IP address. Internet Information Server uses Host Header information when building redirects to the different virtual host names. To use Host Headers, you must provide a host name-to-IP address resolution using either a domain name system (DNS) server or HOSTS files. A HOSTS file provides name resolution for host names to IP addresses.

Note Web browsers that are not HTTP 1.1-compliant can access URLs with Host Headers. Internet Information Server provides non-compliant browsers with a list of servers associated with a given IP address. When the user selects a server, a cookie is placed on the user's disk to direct all future access to the virtual server.

Virtual Server Administration

You can create a virtual server with any Internet Information Server administration tool. Each tool uses a different user interface for creating virtual servers.



The ISM snap-in for Microsoft Management Console uses the New Web Site wizard or New FTP Site wizard to lead you through the virtual server creation process. After you have established the virtual server, you can use the **Virtual Server** property sheets to modify its configuration.

HTLMA uses a Web page to lead you through the creation of the virtual server. You can use this administration tool to establish and modify virtual servers remotely.

With WSH, you can create virtual servers automatically using scripts.

Practice



In this practice, you configure a virtual server by using Windows NT Explorer to create and share the folder with the appropriate permissions. You then use Internet Service Manager to create the virtual server. You can use the optional loopback address to connect to your server to perform these virtual server procedures.

Note The loopback IP address is 127.0.0.1. The loopback address uses loopback drivers to reroute outgoing packets back to the source computer. The loopback drivers allow the packets to bypass the network adapter card completely and return directly to the computer that is performing the action.

To add a WWW virtual server

1. Start Windows NT Explorer.
2. Create the C:\Inetpub\Vroot folder.
3. Copy the contents of C:\Inetpub\Wwwroot to C:\Inetpub\Vroot.
4. Start Internet Service Manager, and then right-click your computer icon.

A context menu appears.

5. Click **New**, and then click **Web Site**.

The New Web Site wizard appears.

6. In the **Description** box, type **copy of wwwroot**
7. Click **Next**.
8. In the **IP Address** box, select your server's IP address or your loopback address (127.0.0.1).

Leave the entry as 80 in the **TCP Port this Web Site should use (Default: 80)** text box.

9. Click **Next**.
10. Type the path to C:\Inetpub\Vroot.
11. Click **Next**.

Only the **Read** and **Script** boxes should be selected.

12. Click **Finish**.
13. Right-click the **Copy of WWWRoot** Web site, and then click **Properties**.

The **Properties** dialog box appears.

14. On the **Web Site** tab, click **Advanced**.
15. Select your IP address.
16. Click **Add**.
17. In the **IP Address** box, select your IP address.
18. In the **TCP Port Of** box, type **80**
19. In the **Host Header Name** box, type *computernameA*

For example, if your computer name is Server1, the Host Header name for your virtual server is Server1A.

20. Click **OK**.
21. In the **Advanced Multiple Web Site Configuration** box, click **OK**.
22. Click **OK** to return to Internet Service Manager.

Note This new virtual server inherits its property sheet settings from the WWW or FTP Services **Master** property sheets.

To start your new Web site

1. Right-click **Copy of WWWRoot**.

A context menu appears.

2. Click **Start**.

To test your virtual server

1. Start Internet Explorer.

2. In the **Address** box, type *computernameA* and then press ENTER.

The Microsoft Internet Information Server home page appears. This verifies that your virtual server is working using host headers.

3. Close Internet Explorer.

Summary

With virtual servers you can host multiple Web and FTP sites on a single computer running Internet Information Server. The Host Headers capability of HTTP 1.1 allows you to associate multiple host names with a single IP address. You can use any Internet Information Server administration tool to create virtual servers.

Review



The following questions are intended to reinforce key information presented in this chapter. If you are unable to answer a question, review the appropriate lesson, and then try the question again.

1. You are interested in changing your Web site configurations pertaining to connections—how many users are connecting, how many users can connect, and how long you tolerate an idle connection before dropping it. Where would you go to adjust these functions?

2. You are working on the beta release of a new product and want to allow the members of your team—and no one else—to access documentation on the intranet. Which property sheet helps you to configure your Web site in this way, and what are the steps you must take to configure your site?

3. You are concerned about users having Write privileges to your Web site. Which property sheet contains this information? Would a user require any special browser functionality in order to write to a Web site?

-
4. After installing Internet Information Server, you configure 10 virtual servers. To verify that the virtual servers are working, you copy the Default.asp file from Inetpub\Wwwroot into the home folder of each virtual server. When you test each of the virtual servers, none of the images on the page are displayed. After troubleshooting, you determine that all of the images are in virtual directories, and these virtual directories are only accessible from the default Web site. How do you make them accessible from all virtual servers?
-
-
-

5. You are the administrator for an intranet at a small accounting firm with 10 employees and 15 computers running Windows NT 4.0. After installing Internet Information Server and configuring three virtual servers, users complain they can only display the default WWW server. What must you do so that the virtual servers are working?
-
-

6. You are the administrator for an intranet at a large accounting firm with 1000 employees and 1500 computers running Windows NT 4.0. After installing Internet Information Server, creating three virtual servers, and installing DNS, users complain they can only display the default WWW server. What must you do so that the clients can display the pages on the virtual servers?
-
-

Last updated January 12, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use



[TechNet Home](#) | [Site Map](#) | [Events](#) | [Downloads](#) | [Personalize](#) | [Worldwide](#) | [Advanced Search](#) |

Chapter 8 - Publishing Information and Applications

Internet Information Server can publish both information and applications. This means your server can contain anything from static pages of information to interactive applications. You can also find and extract information from, and insert information into, databases.

This chapter explains how to:

- Prepare your server and your information for Publishing.
- Install and use interactive applications on your server.
- Publish by using an Open Database Connectivity (ODBC)-compliant data source.

Preparing Information for Publishing

Most Web pages are formatted in HyperText Markup Language (HTML). HTML files are simple ASCII text files with codes embedded to indicate formatting and hypertext links. HTML specifications are changing constantly. You should probably review the HTML specifications (available on the Internet) to fully plan your HTML pages.

Authoring HTML Files

You can use any text editor, such as Notepad or Write, to create and edit your HTML files; but you will probably find an HTML editor, such as Internet Assistant for Microsoft Word, easier to use.

You use the HTML editor or other system to create HTML files, which can include hyperlinks to other files on your system. If you want to include images or sounds, you will also need appropriate software to create and edit those files.

Publishing HTML and Other File Formats

Your files can include images and sound. You can even create links to Microsoft® Office files or to almost any other file format. Remote users must have the correct viewing application to view nonHTML files. For example, if you know that all remote users will have Microsoft Word, you can include links to Microsoft Word .doc files. The user can click the link and the document will appear in Word on the user's computer.

Once you have created your information in HTML or other formats, you can either copy the information to the default directory \Inetsrv\Wwwroot, or you can change the default home directory to the directory containing your information.

MIME Type Configuration

If your server provides files that are in multiple formats, your server must have a Multipurpose Internet Mail Extension (MIME) mapping for each file type. If MIME mapping on the server is not set up for a specific file type, browsers may not be able to retrieve the file. See the Windows NT Registry for the default MIME mappings.

To configure additional MIME mappings start Regedt32.exe and open

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap

Add the value for the MIME mapping required for your server with the following syntax:

```
<mime type>,<filename extension>,<unused parameter>,<gopher type>
```

For example:

```
text/html,html,/unused,1
image/jpeg,jpeg,/unused,5
```

The default entry with the filename extension specified as an asterisk (*) is the default MIME type used when a MIME mapping does not exist. For example, to handle a request for the file Current.vgr when the filename extension .vgr is not mapped to a MIME type, the server will use the MIME type specified for the asterisk extension, which is the type used for binary data. Usually, this will cause browsers to save the file to disk.

Including Other Files with the Include Statement

You can add repetitive information into an HTML file just before sending the file to a user. This feature is handy for including the same text on each HTML page, such as copyright information or a link to the home page.

The format of the include statement is:

```
<!--#include file="value"-->
```

The value must contain the full path, from the home directory of your WWW service.

For example, to include a link to your home page in each HTML document:

1. Create the file linkhome.htm, which contains the HTML codes you want to repeat; for example, a button to your home page. The file would contain HTML that looks similar to this:

```
<A HREF="/homepage.htm"><IMG SRC="/images/button_h.gif"></A>
```

2. Use the filename extension .stm when you create your Web pages (rather than .htm or .html).
3. In each .stm file, use an **include file** statement where you want the repeated information to appear. For example:

```
You can return to: <!--#include file="/linkhome.htm"--> at any time
```

Note that all paths are relative to the WWW home directory and can include virtual roots.

Publishing Dynamic Applications

One of the most exciting features of Microsoft Internet Information Server is the ability to run applications or scripts that remote users start by clicking HTML links or by filling in and sending an HTML form. Using programming languages such as C or Perl, you can create applications or scripts that communicate with the user in dynamic HTML pages.

Creating the Applications or Scripts

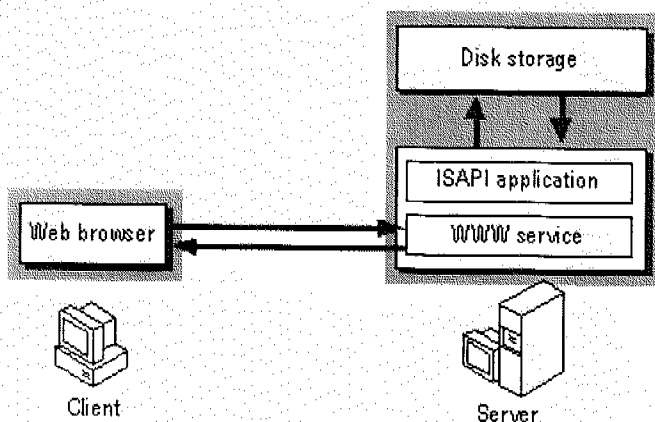
Interactive applications or scripts can be written in almost any 32-bit programming language, such as C or Perl, or as Windows NT batch files (.bat or .cmd). When you write your applications or script you can use one of two supported interfaces, the Microsoft Internet Server Application Programming Interface (ISAPI) or the Common Gateway Interface (CGI). Documentation for ISAPI is available from Microsoft via subscription to the Microsoft Developer Network (MSDN). Documentation for CGI is available on the Internet. Batch files can issue any command valid at the command prompt.

Applications that use ISAPI are compiled as Dynamic Link Libraries (DLLs) that are loaded by the WWW service at startup. Because the programs are resident in memory, ISAPI programs are significantly faster than applications written to the CGI specification.

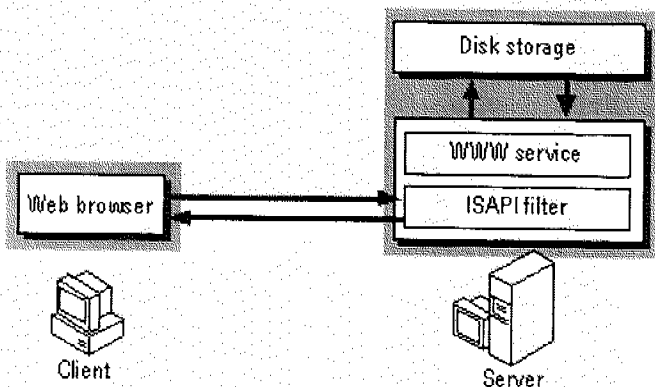
Internet Server API

ISAPI for Windows NT can be used to write applications that Web users can activate by filling out an HTML form or clicking a link in an HTML page on your Web server. The remote application can then take the user supplied information and do almost anything with it that can be programmed, and then return the results in an HTML page or post the information in a database.

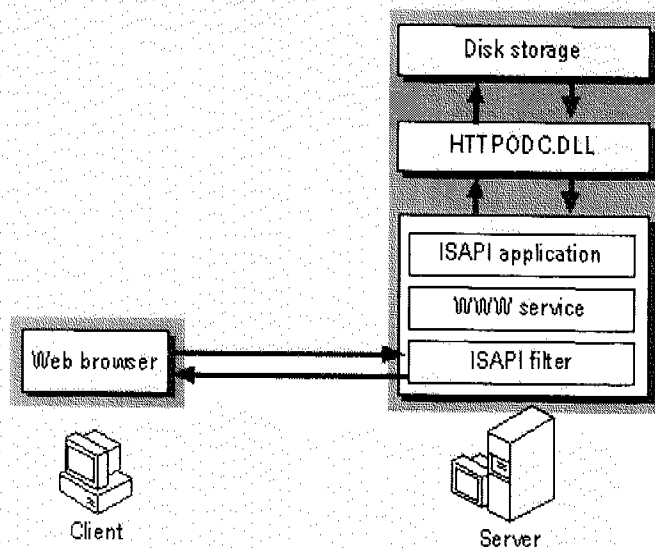
ISAPI can be used to create applications that run as DLLs on your Web server. If you have used Common Gateway Interface (CGI) scripts before, you will find that the ISAPI applications have much better performance because your applications are loaded into memory at server runtime. They require less overhead because each request does not start a separate process.



Another feature of ISAPI allows pre-processing of requests and post-processing of responses, permitting site-specific handling of HyperText Transport Protocol (HTTP) requests and responses. ISAPI filters can be used for applications such as customized authentication, access, or logging.



You can create very complex sites by using both ISAPI filters and applications. ISAPI extensions can also be combined with the Internet Database Connector to create highly interactive sites.

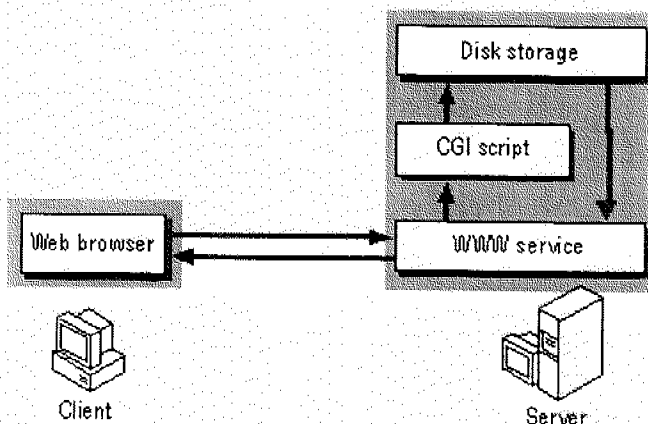


For complete information about programming with ISAPI, see the Microsoft BackOffice Software Development Kit (SDK), available from Microsoft. See the introductory chapter, "Before You Begin," for further information about obtaining the ISAPI SDK.

Common Gateway Interface

The Common Gateway Interface (CGI) is a standard interface used to write applications that remote users can start by filling out an HTML form or clicking a link in an HTML page on your Web server. As with ISAPI, the remote application can then take the user-supplied information and do almost anything that can be programmed, then return the results of the application in an HTML page or post the information in a database. Because simple CGI applications are often written using scripting languages such as Perl, CGI applications are sometimes referred to as "scripts."

Most 32-bit applications that run on Windows NT and conform to the CGI specifications can be used by Microsoft Internet Information Server.



For information about how to convert existing CGI programs or scripts from UNIX®, see Help.

For more information about the CGI specifications, consult the CGI specifications widely available on the Internet.

Installing Your Application on Internet Information Server

Once you have written your application or script, place it in the /Scripts directory, a virtual directory for

applications. This virtual directory has Execute access.

You must also ensure that every process started by your application is running by using an account with adequate permissions. If your application interacts with other files, the account you assign to your program must have the right permissions to use those files. By default, applications run using the IUSR_computername account.

Running Your Application

If your application does not require data from the user, you will typically create a link to your application in a simple HTML file. If your application does require data from the user, you will probably use an HTML form. In other instances you can just send a Uniform Resource Locator (URL), usually containing data parameters, to invoke a program.

An HTML link to a application that does not require input from the user might look like the following example:

<http://www.company.com/scripts/catalog.exe?>

where \Scripts is the virtual directory for interactive applications.

If you are creating a application that requires input from the user, you will need to understand both HTML forms and how to use the forms with ISAPI or CGI. This information is widely available on the Internet or from other sources.

Associating Interpreters with Applications

Because you have the flexibility to create applications in almost any programming language, Internet Information Server uses the filename extension to determine which interpreter to invoke for each application. The default interpreter associations are listed below. You can use the Registry Editor to create additional associations as described in Help.

Extension	Default Interpreter
.exe, .com, .bat, .cmd	Cmd.exe
.idc	Httpodbc.dll

Security Implications

When you allow remote users to run applications on your computer, you run the risk of hackers attempting to break into your system. Microsoft Internet Information Server is configured by default to reduce the risk of malicious intrusion by applications in two important ways.

First, the virtual directory \Scripts contains your applications and is marked as a application directory. Only an administrator can add programs to a directory marked as an application directory. Thus, unauthorized users cannot copy a malicious application and then run it on your computer without first gaining administrator access.

Second, if you have configured the WWW service to allow only anonymous logons, all requests from remote users will use the IUSR_computername account. By default, the IUSR_computername account is unable to delete or change files by using the Windows NT File System (NTFS) unless specifically granted access by an administrator. Thus, even if a malicious program were copied to your computer, it would be unable to cause much damage to your content because it will only have IUSR_computername access to your computer and files.

Publishing Information and Using a Database

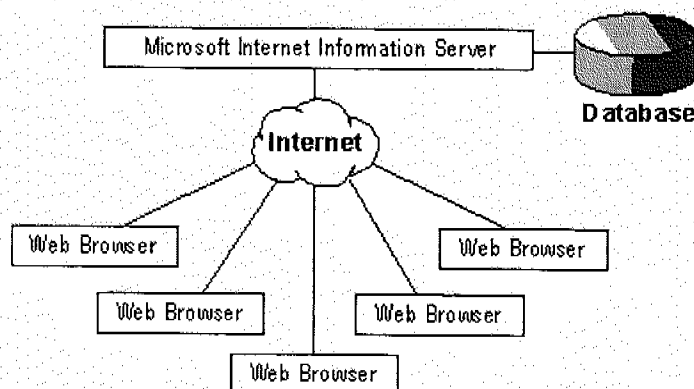
With the WWW service and the Open Data Base Connectivity (ODBC) drivers provided with Internet Information Server, you can:

- Create Web pages with information contained in a database.

- Insert, update, and delete information in the database based on user input from a Web page.
- Perform other Structured Query Language (SQL) commands.

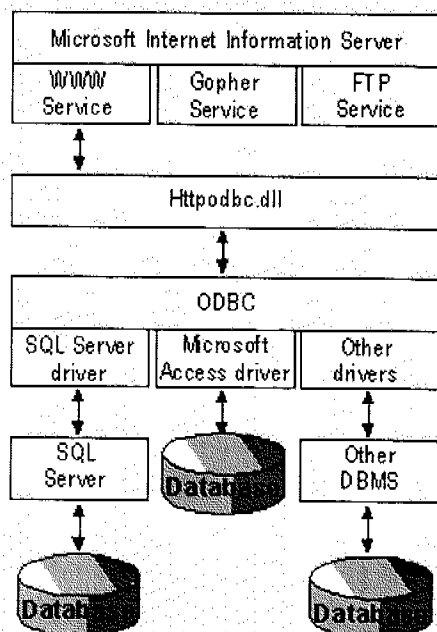
How the Internet Database Connector Works

Conceptually, database access is performed by Internet Information Server as shown in the following diagram.



Web browsers (such as Internet Explorer, or browsers from other companies such as Netscape) submit requests to the Internet server by using HTTP. The Internet server responds with a document formatted in HTML. Access to databases is accomplished through a component of Internet Information Server called the Internet Database Connector. The Internet Database Connector, `Httpodbc.dll`, is an ISAPI DLL that uses ODBC to gain access to databases.

The following illustration shows the components for connecting to databases from Internet Information Server.



`Httpodbc.dll` uses two types of files to control how the database is accessed and how the output Web page is constructed. These files are Internet Database Connector (.idc) files and HTML extension (.htx) files.

The Internet Database Connector files contain the necessary information to connect to the appropriate ODBC data source and execute the SQL statement. An Internet Database Connector file also contains the

name and location of the HTML extension file.

The HTML extension file is the template for the actual HTML document that will be returned to the Web browser after the database information has been merged into it by Httpodbc.dll.

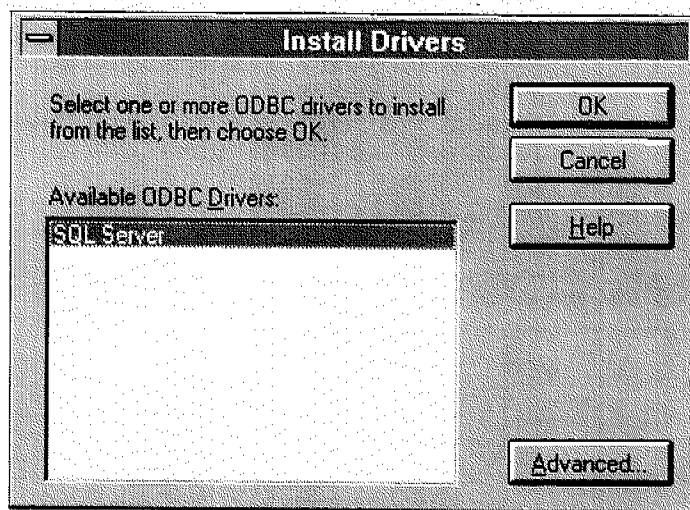
Installing ODBC and Creating System Data Sources

When the ODBC option is selected during setup, ODBC version 2.5 components are installed. This version of ODBC supports System DSNs (Data Source Names) and is required for using ODBC with Microsoft Internet Information Server.

System DSNs were introduced in ODBC version 2.5 to allow Windows NT services to use ODBC.

[cchev] To install the ODBC drivers

1. If you did not install the ODBC Drivers and Administration option, run Setup again by double-clicking the Internet Information Server Setup icon in the Microsoft Internet Server program group of Program Manager. You will need the Internet Information Server compact disc, or a network installation directory containing the complete contents of the compact disc.
2. Choose the OK button.
3. Choose the Add/Remove button.
4. Choose the OK button.
5. Select the ODBC Drivers and Administration option.
6. Choose the OK button.
7. The Install Drivers dialog box appears.



8. To install the SQL Server driver, select the SQL Server driver from the Available ODBC Drivers list box, and choose the OK button.

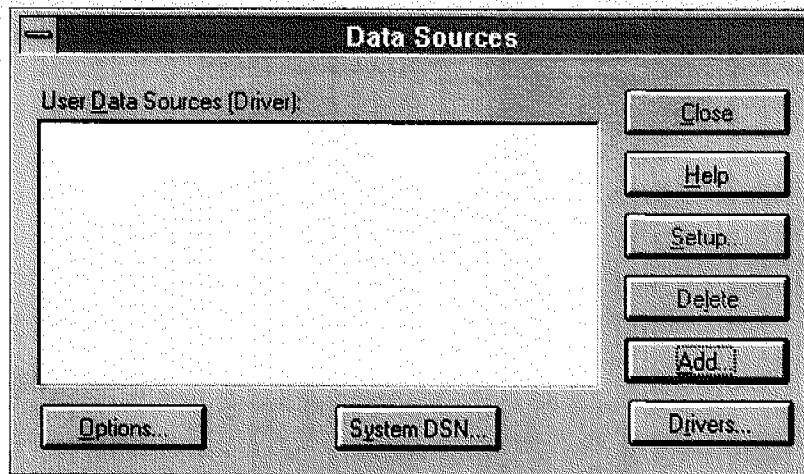
Setup completes copying files.

[cchev] To create the system data sources

1. Double-click the Control Panel icon in the Main program group of Program Manager.
2. Double-click the ODBC icon.

The ODBC Data Sources dialog box appears.

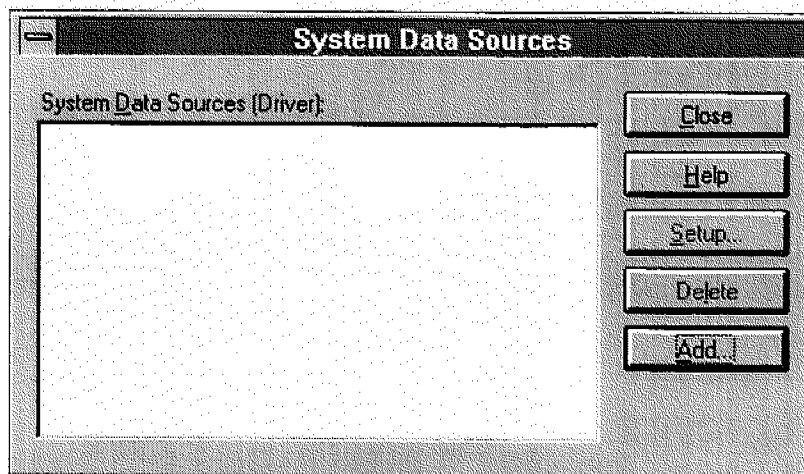
You may see other data sources in the list if you previously installed other ODBC drivers.



3. Choose the System DSN button.

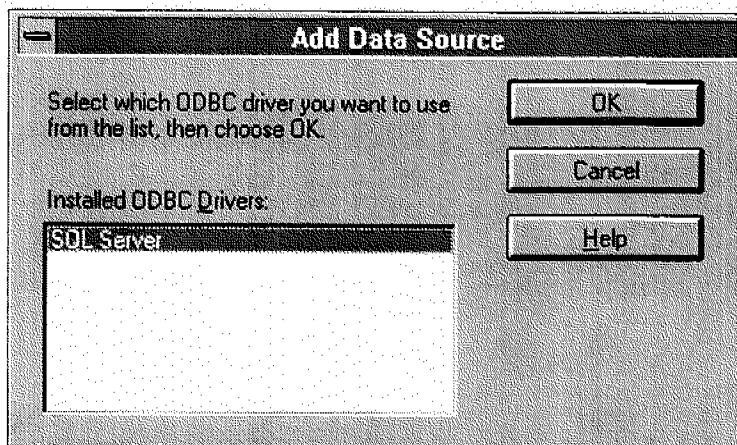
Important Be sure to click the System DSN button. The Internet Database Connector will work only with System DSNs.

The System Data Sources dialog box appears.

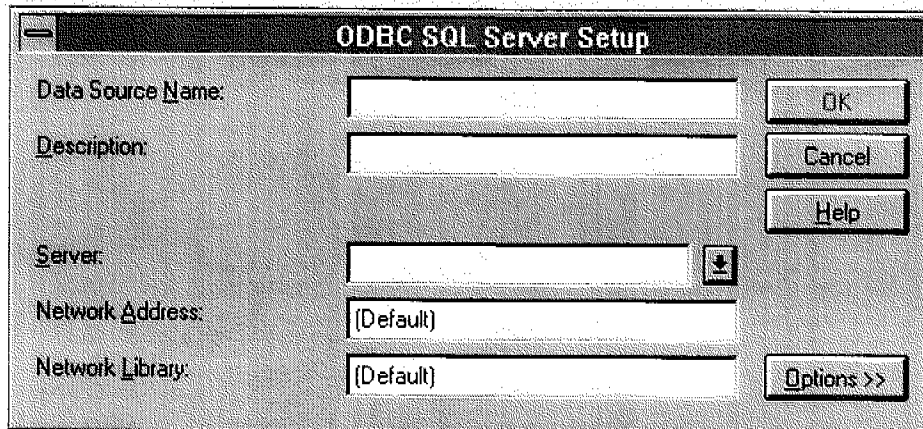


4. Choose the Add button.

The Add Data Source dialog box appears.



5. Select SQL Server from the list box and click OK. The ODBC SQL Server Setup dialog box will appear.



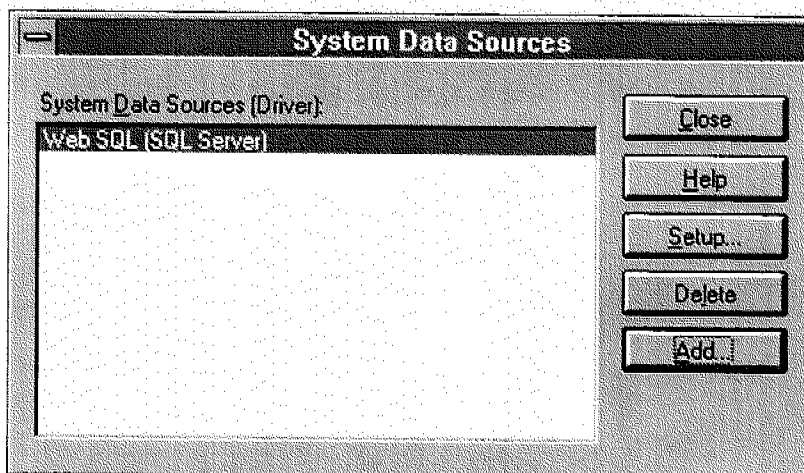
6. Enter the name of the data source.

The data source name is a logical name used by ODBC to refer to the SQL Server driver and the actual server name on which SQL Server is running. You can also use the server name "(local)" if SQL Server is running on this computer. The data source name is used in Internet Database Connector files to tell Internet Information Server where to access the data.

The server name, network address, and network library are specific to your installation. If you do not know what to enter in these controls, accept the defaults. To find out the details, click the Help button and find the section that describes your network.

7. Choose the OK button.

The System Data Sources dialog box will be displayed again, but now will have the name of the data source displayed.



8. Choose the Close button to close the System Data Sources dialog box.
9. Choose the Close button to close the Data Sources dialog box.
10. Choose the OK button to complete the ODBC and DSN setup.

Authoring Web Pages with Database Access

In order to provide access to a SQL database from your Web page, you will need to create an Internet Database Connector file and an HTML extension file.

For more information on creating these files, see Help.

Last updated January 10, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use...
